Distributed adaptive control of networked cooperative mobile manipulators

Alessandro Marino

***Abstract*—This paper deals with the networked control of loosely or tightly connected cooperative manipulators in charge of achieving a cooperative task, that is specified by means of proper task-oriented variables depending on the full state of the system. Since the full state is not known to robots, a two-layer architecture is designed. At the first level, each arm controller runs a distributed observer that estimates the system state. At the second level, this estimation is adopted to compute the local control input as in the case of a centralized solution is available. In addition, since the dynamic parameters of the arms might not be perfectly known, the local control law is made adaptive in order to counteract this uncertainty. The designed solution is suitable not only for pure motion coordination tasks but can be exploited also in those cases where a closed kinematic chain is generated by multi-robot object manipulation. In this situation, the objective is to both move the object and limit the internal stresses on it, that, however, cannot be locally computed. To overcome this issue, the wrench exerted by each robot on the object is decomposed in an external component (contributing to the motion of the object) and in a internal component that is locally estimated and, then, regulated. The approach was validated by simulation with 6-DOFs serial chain manipulators mounted on a mobile platform and performing cooperative tasks.**

***Keywords*—Networked Robots; Distributed Control; Cooperative Robots.**

I. Introduction

Two of the key points behind the strategic initiative of Industry 4.0 concerning next generation smart factories are high flexibility and reconfigurability which are believed to make more profitable the production of small-lot and, then, customizable products [1]. In fact, such features minimize the setup time and, then, make possible to easily switch between one product to an other. However, industrial robots, despite they play a key role for the automation of production processes, are currently difficult and time consuming to set-up and program. This process is even more complex if two or more robots need to loosely or tightly cooperate for achieving tasks. It is clear, then, that highly reconfigurable manipulation systems require to efficiently connect, disconnect and reconnect one or more units to the overall system, preferably within a decentralized architecture in which a central coordination unit is not present (eg., in aerial or underwater cooperative manipulation) [2].

This paper presents a decentralized architecture that allows to accomplish complex manipulation tasks in which robots only rely on locally available information. Differently from existing works, both the cases of pure kinematic control and interaction control are taken into account within the same framework with only slight modification of the control law. The former might be useful for the coordination of loosely coupled robots or for handling deformable and flexible objects; while the latter is necessary for cooperatively handling rigid objects and avoid internal stresses [3]. The cooperative motion task is expressed by proper meaningful task variables, whose value remarkably depends on the end-effector configurations of all robots in the team. This variable might be useful to specify the centroid of the end-effector formation, the position of the object or the end-effector relative configurations, for example. A desired value of the task variables is known only by a subset of the robots, while the other robots estimate this reference via explicit communication. The structure of the task variables makes the control input of each arm depend on the state of the overall system, leading, in general, to the necessity of a central unit. By building on the results in [4], a layered architecture is adopted to achieve the same result in a distributed way. At the first level, each manipulator continuously updates an estimate of the overall state of the system by adopting a consensus-based observer. At the second level, such an estimate is used to compute the adaptive control input necessary to achieve the cooperative task.

Concerning the case of robots tightly connected for the manipulation of a common rigid object, internal generalized forces that do not contribute to the motion of the object might raise, especially in the case a centralized controller does not exist. It is well known that the computation of the internal forces requires global information as well, nevertheless, in this paper, they are estimated in a distributed fashion and controlled by properly building on the solution designed for the pure kinematic coordination.

The paper is organized as follows. Section II describes research work related to distributed coordination of Euler-Lagrangian systems and distributed cooperative manipulation. In Section III, the problem description and mathematical framework are introduced. A solution to the pure kinematic coordination problem is given in Section IV, while in Section VI this solution is extended in order to take into consideration also interaction forces. Simulation results in Section VII show that both the pure kinematic and the interaction control can be handled by the designed framework in the case of 4 manipulators mounted on a mobile base. Finally, conclusions are drawn and future work presented in Section VIII.

II. Related Works

In the last decades there has been an increasing interest in the formulation of decentralized paradigms for multi-arm systems thanks to their potential advantages for modern manufacturing or, in general, for field robotics. However, the control of such systems is made difficult by the fact that each robot has access only to local information while global coordination might be required by the task. For example, in [5] the contraction theory is exploited to synchronize in joint space Lagrangian systems, and the proposed control law

University of Salerno, Via Giovanni Paolo II, 132, 84084, Salerno (SA), Italy email: {almarino}@unisa.it

requires only local position/velocity coupling feedback. Recurrent neural networks are exploited in [6] for the kinematic control of cooperative redundant manipulator. The proposed neural network is composed of independent modules, each of them controlling a single manipulator and all of them together generate an emergent collective behavior to complete the cooperative task. With regards to synchronization of rigid-link manipulators in task-space, a solution that exploits passivity and takes into account both dynamic uncertainties and time-varying communication delays is developed in [7] and [8]. Remarkably, in the latter work also the uncertainty in the knowledge of kinematics parameters is considered.

Concerning the case of tightly connected arms for the manipulation of a common object, the problem has been widely faced since the 80's in the framework of centralized architecture (see [9] and references therein).

Decentralized architectures for common load manipulation have been proposed as well. For example, in [10] a fully decentralized motion and force control is proposed for a multi-robot system rigidly transporting a load on a plane, in which there is no explicit communication between robots. In [11], a pushing leader and multiple followers are employed to transport an object on a plane. Followers estimate leader's trajectory and control internal forces to ensure a stable grasping. A decentralized method for collective transport of an object on a plane without rigid grasping, and by simple agents with limited capabilities is described in [12]. In [13], an impedance based leader-following approach is presented for a dual-arm setup: each robot uses only its own force, position and velocity measurements, while explicit communication between mobile arms is not required. In [14], a flexible payload transported by planar robots is considered. A desired constant velocity is set for the object and the load transportation problem is faced in a similar fashion to the formation-control problem. A leader-follower approach is designed in [15] for the cooperative transportation of an object moving on a plane. Followers synchronize to the motion applied by the leader to the object without using explicit communication. Similarly in [16], the leader navigates towards the goal, while a single follower helps in transporting a load. Both robots have to ensure that the transported object does not collide with obstructions. The strategy is not generalized to any number of robots and interaction forces are not considered. Authors in [17] devised a decentralized motion control strategy for moving a load constrained on a plane in the case of uncertain parameters of the load. A robust control based on some assumptions is devised and, also in this case, closed chain constraints are not considered. With respect to existing works, this paper

- deals with both pure kinematic and force control tasks within the same observer-controller schema;
- allows to estimate and control squeezing forces in a decentralized fashion;
- considers robots and objects moving in the 3D space dealing with both position and orientation of the robots and the manipulated object;
- adopts a robust strategy for controlling squeezing forces and counteract the lack of information typical in decentralized control strategy;
- allows each robot to estimate the full state of the system, that might be useful to achieve complex control strategy as in the case of centralized control.

III. MATHEMATICAL BACKGROUND

A. List of symbols

In Table I the main symbols used across the paper are reported.

B. System modeling

Let us consider a work-cell composed by N serial-chain manipulators eventually mounted on a mobile base. The dynamic model of the ith mobile manipulator can be written according to the Newton-Lagrange formulation

$$\begin{aligned}\boldsymbol{M}_i(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_i+\boldsymbol{C}_i(\boldsymbol{q}_i,\dot{\boldsymbol{q}}_i)\dot{\boldsymbol{q}}_i+\boldsymbol{F}_i\dot{\boldsymbol{q}}_i+\boldsymbol{g}_i(\boldsymbol{q}_i) &= \boldsymbol{Y}_i(\boldsymbol{q}_i,\dot{\boldsymbol{q}}_i,\ddot{\boldsymbol{q}}_i)\boldsymbol{\pi}_i \\ &= \boldsymbol{\tau}_i - \boldsymbol{J}_i^{\mathrm{T}}(\boldsymbol{q}_i)\boldsymbol{h}_i, \quad (1)\end{aligned}$$

N	Number of robots
$\boldsymbol{x}_i \in \mathbb{R}^p$	End-effector configuration of the ith manipulator
$\boldsymbol{q}_i \in \mathbb{R}^{n_i}$	Joint configuration of the ith manipulator
$\boldsymbol{\tau}_i \in \mathbb{R}^{n_i}$	Torque input of the ith manipulator
$\boldsymbol{\pi}_i \in \mathbb{R}^{n_{\pi_i}}$	Vector of dynamic parameters for the ith robot
$\boldsymbol{v}_i$	Generic variable $\boldsymbol{v}$ relative to the i-th agent
$\boldsymbol{v}$	Stacked vector collecting the variables $\boldsymbol{v}_i$ for all the agents, i.e., $\boldsymbol{v} = \begin{bmatrix}\boldsymbol{v}_1^{\mathrm{T}} & \boldsymbol{v}_2^{\mathrm{T}} & \dots & \boldsymbol{v}_N^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$
${}^i\hat{\boldsymbol{v}}$	Estimate of $\boldsymbol{v}$ performed by i-th agent
$\hat{\boldsymbol{v}}^\star$	Collection of the estimation vectors of the different agents; i.e., $\hat{\boldsymbol{v}}^\star = \begin{bmatrix}{}^1\hat{\boldsymbol{v}}^{\mathrm{T}} & {}^2\hat{\boldsymbol{v}}^{\mathrm{T}} & \dots & {}^N\hat{\boldsymbol{v}}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$
${}^i\tilde{\boldsymbol{v}}$	Estimation error performed by i-th agent; i.e., ${}^i\tilde{\boldsymbol{v}} = \boldsymbol{v} - {}^i\hat{\boldsymbol{v}}$
$\tilde{\boldsymbol{v}}^\star$	Collection of the estimation errors of the different agents; i.e., $\tilde{\boldsymbol{v}}^\star = \begin{bmatrix}{}^1\tilde{\boldsymbol{v}}^{\mathrm{T}} & {}^2\tilde{\boldsymbol{v}}^{\mathrm{T}} & \dots & {}^N\tilde{\boldsymbol{v}}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$
$\boldsymbol{\sigma}(\boldsymbol{x}) \in \mathbb{R}^m$	Cooperative state dependent task variable
$\boldsymbol{\sigma}_d \in \mathbb{R}^m$	Desired value of the cooperative task variable
$\tilde{\boldsymbol{\sigma}} \in \mathbb{R}^m$	Task error $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}_d - \boldsymbol{\sigma}$
$\boldsymbol{\zeta}_d \in \mathbb{R}^m$	Reference signal such as $\boldsymbol{\zeta}_d = \dot{\boldsymbol{\sigma}}_d + k_\sigma \boldsymbol{\sigma}_d$
$\boldsymbol{\gamma} \in \mathbb{R}^m$	$\boldsymbol{\gamma} = \boldsymbol{\zeta}_d - k_\sigma \boldsymbol{\sigma}(\boldsymbol{x}) = \dot{\boldsymbol{\sigma}}_d + k_\sigma(\boldsymbol{\sigma}_d - \boldsymbol{\sigma})$
${}^i\hat{\boldsymbol{\gamma}} \in \mathbb{R}^m$	Estimate of $\boldsymbol{\gamma}$ made by robot i and defined as ${}^i\hat{\boldsymbol{\gamma}} = {}^i\hat{\boldsymbol{\zeta}} - k_\sigma \boldsymbol{\sigma}({}^i\hat{\boldsymbol{x}})$
$\boldsymbol{R}_{m\times n}$	Generic matrix of dimension $m \times n$
$\boldsymbol{I}_{m\times m} = \boldsymbol{I}_m$	Identity matrix of dimension $m \times m$
$\boldsymbol{O}_{m\times m} = \boldsymbol{O}_m$	Null matrix of dimension $m \times m$
$\boldsymbol{1}_m$ ($\boldsymbol{0}_m$)	Columns vector of dimension $m \times 1$ with all elements equal to 1 (0)

TABLE I: Table listing the main variables adopted in the paper.

where $\boldsymbol{q}_i \in \mathbb{R}^{n_i}$ ($\dot{\boldsymbol{q}}_i$, $\ddot{\boldsymbol{q}}_i$) is the joint position (velocity, acceleration) vector, $\boldsymbol{\tau}_i \in \mathbb{R}^{n_i}$ is the joint torque vector, $\boldsymbol{M}_i(\boldsymbol{q}_i) \in \mathbb{R}^{n_i \times n_i}$ is the symmetric positive definite inertia matrix, $\boldsymbol{C}_i(\boldsymbol{q}_i, \dot{\boldsymbol{q}}_i) \in \mathbb{R}^{n_i \times n_i}$ is the centrifugal and Coriolis terms matrix, $\boldsymbol{F}_i \in \mathbb{R}^{n_i \times n_i}$ is the matrix modeling viscous friction, $\boldsymbol{g}_i(\boldsymbol{q}_i) \in \mathbb{R}^{n_i}$ is the vector of gravity terms, and $\boldsymbol{h}_i \in \mathbb{R}^p$ is the vector of interaction forces between the robot's end-effector and the environment. Moreover, $\boldsymbol{Y}_i(\boldsymbol{q}_i, \dot{\boldsymbol{q}}_i, \ddot{\boldsymbol{q}}_i) \in \mathbb{R}^{n_i \times n_{\pi_i}}$ is the known regressor matrix uniformly continuous in its arguments, and $\boldsymbol{\pi}_i \in \mathbb{R}^{n_{\pi_i}}$ is the constant vector of the dynamic parameters of the arm. Concerning $\boldsymbol{J}_i(\boldsymbol{q}_i) \in \mathbb{R}^{p \times n_i}$, the end-effector configuration of the i th manipulator with respect to the world frame Σ_k is denoted by $\boldsymbol{x}_i \in \mathbb{R}^p$ with

$$\boldsymbol{x}_i = \boldsymbol{k}(\boldsymbol{q}_i), \tag{2}$$

where $\boldsymbol{k}_i(\cdot) : \mathbb{R}^{n_i} \to \mathbb{R}^p$, with $n_i \geq p$, is the direct kinematic function mapping between the joint space and the task space, and

$$\dot{\boldsymbol{x}}_i = \boldsymbol{J}_i(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i \tag{3}$$

represents the differential kinematics, being $\boldsymbol{J}_i(\boldsymbol{q}_i) = \dfrac{\partial \boldsymbol{k}(\boldsymbol{q}_i)}{\partial \boldsymbol{q}_i}$ the analytic Jacobian matrix of the i th arm supposed to be known. In the following, for the sake of compactness the dependence of $\boldsymbol{J}_i(\boldsymbol{q}_i)$ on $\boldsymbol{q}_i$ will be omitted, i.e., $\boldsymbol{J}_i = \boldsymbol{J}(\boldsymbol{q}_i)$. It is supposed that only an estimation of the model in (1) is available:

$$\hat{\boldsymbol{M}}_i(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_i + \hat{\boldsymbol{C}}_i(\boldsymbol{q}_i, \dot{\boldsymbol{q}}_i)\dot{\boldsymbol{q}}_i + \hat{\boldsymbol{F}}_i\dot{\boldsymbol{q}}_i + \hat{\boldsymbol{g}}_i(\boldsymbol{q}_i) = \boldsymbol{Y}_i(\boldsymbol{q}_i, \dot{\boldsymbol{q}}_i, \ddot{\boldsymbol{q}}_i)\hat{\boldsymbol{\pi}}_i, \tag{4}$$

where the symbol $\hat{\cdot}$ denotes the estimate of the corresponding quantity.
The following vectors and matrices are defined according to the notation in Table I:

$$\boldsymbol{x} = \left[\boldsymbol{x}_1^{\mathrm{T}}, \quad \boldsymbol{x}_2^{\mathrm{T}}, \quad \ldots, \quad \boldsymbol{x}_N^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{pN}, \tag{5}$$

$$\boldsymbol{q} = \left[\boldsymbol{q}_1^{\mathrm{T}}, \quad \boldsymbol{q}_2^{\mathrm{T}}, \quad \ldots, \quad \boldsymbol{q}_N^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{n}, \tag{6}$$

and

$$\boldsymbol{J} = \boldsymbol{J}(\boldsymbol{q}) = \mathrm{diag}\left\{\boldsymbol{J}_1, \boldsymbol{J}_2, \ldots, \boldsymbol{J}_N\right\} \in \mathbb{R}^{Np \times n}, \tag{7}$$

with $n = \sum_{i=1}^{N} n_i$.
Based on (3)-(7), it is straightforward to see that $\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}$.

Remark 3.1: From the above formulation, it can be noticed that it is not required for the lagrangian systems to have the same dynamics. It is possible for them to have different models and a different number of degrees of mobilities n_i.

The following assumptions will be made in the following.

Assumption 3.1: There exists a positive scalar constant α such that $\|\boldsymbol{J}_i(\boldsymbol{q}_i)\| \leq \alpha < \infty, \forall\, \boldsymbol{q}_i$ $(i = 1, 2, \ldots, N)$.

Assumption 3.2: The Jacobian matrix $\boldsymbol{J}_i(\boldsymbol{q}_i)$ is such as $\lambda_{min}(\boldsymbol{J}_i(\boldsymbol{q}_i)\boldsymbol{J}_i(\boldsymbol{q}_i)^{\mathrm{T}}) > 0$ $(i = 1, 2, \ldots, N)$ along the robot trajectories, where $\lambda_{min}(\cdot)$ ($\lambda_{max}(\cdot)$) is the smallest (largest) eigenvalue of the corresponding matrix argument.

Assumption 3.3: Each robot is equipped with a force sensor to measure the generalized interaction forces $\boldsymbol{h}_i$ in (1).

Assumption 3.1 imposes a smoothness constraint on the direct kinematic function that is always verified for some constant α, Assumption 3.2 ensures that the Jacobian matrix has full rank and is away from singularities.

C. Object dynamics and force decomposition

Let us consider a rigid object rigidly grasped by the N robots. The object configuration is denoted by $\boldsymbol{x}_o \in \mathbb{R}^p$ and its dynamics can be generally described by

$$\boldsymbol{M}_o\ddot{\boldsymbol{x}}_o + \boldsymbol{C}_o(\boldsymbol{x}_o, \dot{\boldsymbol{x}}_o) + \boldsymbol{g}_o(\boldsymbol{x}_o) = \boldsymbol{h}_o, \tag{8}$$

where $\boldsymbol{M}_o \in \mathbb{R}^{p \times p}$ is the inertia matrix, $\boldsymbol{C}_o \in \mathbb{R}^{p \times p}$ represents the Coriolis and centripetal terms, $\boldsymbol{g}_o \in \mathbb{R}^p$ the gravitational forces, and $\boldsymbol{h}_o \in \mathbb{R}^p$ is the resultant of the generalized forces exerted by the manipulators on the object, such as:

$$\boldsymbol{h}_o(x) = \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{x}_o)\boldsymbol{h}, \tag{9}$$

where $\boldsymbol{h} \in \mathbb{R}^{Np}$ is the collective vector of the generalized forces at the manipulator end-effectors:

$$\boldsymbol{h} = \left[\boldsymbol{h}_1^{\mathrm{T}}, \quad \boldsymbol{h}_2^{\mathrm{T}}, \quad \ldots, \quad \boldsymbol{h}_N^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{Np}, \tag{10}$$

and $\boldsymbol{G} = [\boldsymbol{G}_1 \quad \ldots \quad \boldsymbol{G}_N] \in \mathbb{R}^{p \times Np}$ is the well-known grasp matrix with

$$\boldsymbol{G}_i = \boldsymbol{G}_i(\boldsymbol{r}_i); \tag{11}$$

being $\boldsymbol{r}_i$ the vector pointing from the object position $\boldsymbol{p}_o$ to the position of the i th end-effector (the so called virtual stick). By assuming that the kinematics of each robot is referred to $\boldsymbol{p}_o$, $\boldsymbol{G}_i$ ($\forall\, i$) becomes independent on the vector $\boldsymbol{r}_i$ [3], i.e.,

$$\boldsymbol{G}_i = \boldsymbol{I}_p. \tag{12}$$

The following assumption is made.

Assumption 3.4: Each manipulator knows the geometry and the dynamic parameters of the object to carry and refers its kinematics to the same point $\boldsymbol{p}_o$.

The above assumption is not a strong one; in fact, the strategy devised in [18] can be adopted in the case the object is not known beforehand.

D. Information exchange

As usual in the distributed control, the information exchange between the robots is described by a connectivity graph $\mathcal{G}(\mathcal{E}, \mathcal{V})$ characterized by its topology (see [19]), i.e., the set $\mathcal{V}$ of the indexes labeling the N vertices (nodes), the set of edges (arcs) $\mathcal{E} = \mathcal{V} \times \mathcal{V}$ connecting the nodes, and the $(N \times N)$ Adjacency matrix

$$\boldsymbol{A} = \{a_{ij}\} : \quad a_{ii} = 0, \quad a_{ij} = \begin{cases} 1 & \text{if } (j, i) \in \mathcal{E} \\ 0 & \text{otherwise,} \end{cases}$$

whose element a_{ij} is different from zero if node j can send information to node i. Moreover, we assume that the i th robot receives information only from a reduced set of nodes (called its neighbors) $\mathcal{N}_i = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}\}$, and it does not know the topology of the overall communication graph.
We assume the graph is connected and *undirected*. In addition

to the Adjacency matrix, the $(N \times N)$ Laplacian matrix defined as

$$\boldsymbol{L} = \{l_{ij}\} : \quad l_{ii} = \sum_{j=1}^{N} a_{ij}, \quad l_{ij} = -a_{ij},\ i \neq j$$

is commonly used. Concerning this matrix, zero is always an eigenvalue with the $(N \times 1)$ vector of all ones, $\mathbf{1}_N \in \mathbb{R}^N$, as the corresponding right eigenvector, i.e., $\boldsymbol{L}\mathbf{1}_N = \boldsymbol{O}_N$, with $\boldsymbol{O}_N$ the $(N \times 1)$ null vector. Hence, $\text{rank}(\boldsymbol{L}) \leq N - 1$ where the equality holds when the graph is strongly connected [19].

IV. Distributed observer-control scheme

In this section the problem of distributed task-oriented control of cooperative manipulators is presented together with the proposed solution.

A. Cooperative task and control objective

Let us consider the general case of a work-cell task coded by a task function $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{x}) \in \mathbb{R}^m$ that is function of the overall state $\boldsymbol{x}$ of the cell. The task has the following expression:

$$\boldsymbol{\sigma} = \boldsymbol{J}_\sigma \boldsymbol{x}, \quad \dot{\boldsymbol{\sigma}} = \boldsymbol{J}_\sigma \dot{\boldsymbol{x}}, \quad \ddot{\boldsymbol{\sigma}} = \boldsymbol{J}_\sigma \ddot{\boldsymbol{x}} \tag{13}$$

where $\boldsymbol{J}_\sigma \in \mathbb{R}^{m \times pN}$ is the constant task Jacobian matrix. The user assigns the desired value of the task function, namely $\boldsymbol{\sigma}_d(t)$, and the objective is to compute the torque input vectors $\boldsymbol{\tau}_i$ $(i = 1, 2, \ldots, N)$ in (1) such as $\boldsymbol{\sigma}(\boldsymbol{x})$ asymptotically tracks $\boldsymbol{\sigma}_d$ *in absence* of a central controller and in the case of manipulator dynamics not perfectly known as in (4).

Assumption 4.1: The desired trajectory of the cooperative task function $\boldsymbol{\sigma}_d(t)$ is known to only a subset of the robots in the team and needs to be estimated by the other ones.

Remark 4.1: Centralized solution to the problem described here exists and an example is in [20], where the case of two manipulators performing a cooperative task in the case of known dynamics is tackled. Moreover, distributed solution were proposed in [21], but vehicles with single integrator dynamics were considered. In this paper, the solution to this problem (even for the pure kinematic control case) is made difficult by the uncertain manipulators' dynamics, and by the fact that, while the control input of the i th arm depends on the full work-cell state $\boldsymbol{x}$ (as the task function in (13)), each arm has only access to local information from its on-board sensors and information coming from its neighbours according to the communication graph $\mathcal{G}$ (see Section III-D).

In the following, the solution to the problem described above is presented. The designed strategy consists in a two layers schema. At the first level, each arm interacts with a subset of the arms in the cell by exchanging information in order to estimate the task reference trajectory $\boldsymbol{\sigma}_d$ and the overall state of the system $\boldsymbol{x}$. At the second level this estimate is used within an adaptive control law whose aim is to counteract the uncertainties and achieve the global task.

B. Distributed estimation of the task reference $\boldsymbol{\sigma}_d(t)$

Because of Assumption 4.1 and differently from most of the work in literature, it is assumed that the cooperative task function reference trajectory $\boldsymbol{\sigma}_d$ is only known to a subset of the robots in the team. Moreover, it is assumed that this trajectory is three times differentiable with bounded third order derivative. This is not a strong assumption, since high order regular trajectories improve the path tracking capabilities of robot manipulators while preserving mechanical transmissions [22].
Therefore, it is needed to estimate this reference in accordance to the communication topology and in order to compute the control torque as shown in the following sections. To the scope, the approach in [23] is exploited as summarized below for the reader's convenience.
Let $\boldsymbol{\zeta}_d$ be defined such as $\boldsymbol{\zeta}_d = \dot{\boldsymbol{\sigma}}_d + k_\sigma \boldsymbol{\sigma}_d$, whose derivative is $\dot{\boldsymbol{\zeta}}_d = \ddot{\boldsymbol{\sigma}}_d + k_\sigma \dot{\boldsymbol{\sigma}}_d$ and where k_σ is a positive scalar gain. Moreover, let ${}^i\hat{\boldsymbol{\zeta}}$ be the estimation of $\boldsymbol{\zeta}_d$ made by robot i, whose dynamics is

$$\begin{bmatrix} {}^i\dot{\hat{\boldsymbol{\zeta}}} \\ {}^i\ddot{\hat{\boldsymbol{\zeta}}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{O}_m & \boldsymbol{I}_m \\ \boldsymbol{O}_m & \boldsymbol{O}_m \end{bmatrix} \begin{bmatrix} {}^i\hat{\boldsymbol{\zeta}} \\ {}^i\dot{\hat{\boldsymbol{\zeta}}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{O}_m \\ \boldsymbol{I}_m \end{bmatrix} \boldsymbol{u}_{\zeta,i}, \tag{14}$$

where $\boldsymbol{O}_m$ and $\boldsymbol{I}_m$ are the null and identity matrices in $\mathbb{R}^{m \times m}$, respectively, and with $\boldsymbol{u}_{\zeta,i} \in \mathbb{R}^m$ chosen as

$$\begin{aligned} \boldsymbol{u}_{\zeta,i} = k_1 \text{sign}\left[\sum_{j \in \mathcal{N}_i}^{N} ({}^j\dot{\hat{\boldsymbol{\zeta}}} - {}^i\dot{\hat{\boldsymbol{\zeta}}}) + b_i(\dot{\boldsymbol{\zeta}}_d - {}^i\dot{\hat{\boldsymbol{\zeta}}})\right. \\ \left. + k_2 \text{sig}\left[\sum_{j \in \mathcal{N}_i}^{N} ({}^j\hat{\boldsymbol{\zeta}} - {}^i\hat{\boldsymbol{\zeta}}) + b_i(\boldsymbol{\zeta}_d - {}^i\hat{\boldsymbol{\zeta}})\right]^{0.5}\right]; \end{aligned} \tag{15}$$

where k_1 and k_2 are positive scalar gains, $\text{sign}(\cdot)$ is the component-wise *signum* function, while $\text{sig}(\cdot)$ is such as

$$\text{sig}(\boldsymbol{x})^{0.5} = \left[\text{sign}(x_1)|x_1|^{0.5}\ \ \text{sign}(x_2)|x_2|^{0.5} \ldots\ \text{sign}(x_n)|x_n|^{0.5}\right]^{\mathrm{T}}.$$

Concerning b_i in (15), it is equal to 1 if robot i knows the trajectory $\boldsymbol{\sigma}_d(t)$ and is 0 otherwise.
The following theorem holds.

Theorem 4.2: Let us consider the system in (14) with update law in (15), the estimation error ${}^i\tilde{\boldsymbol{\zeta}} = \boldsymbol{\zeta}_d - {}^i\hat{\boldsymbol{\zeta}}$ (${}^i\dot{\tilde{\boldsymbol{\zeta}}} = \dot{\boldsymbol{\zeta}}_d - {}^i\dot{\hat{\boldsymbol{\zeta}}}$) converges to the origin in finite time $T_s > 0$, provided that the communication graph as in Section III-D is undirected and connected, that gains k_1 and k_2 are properly chosen and that b_i is 1 for at least one value of i (i.e., at least one robot knows the reference $\boldsymbol{\sigma}_d(t)$ and its derivatives).

Proof: The proof is based on the same arguments as in [23] and is, then, omitted here. The same theorem suggests how to choose gains k_1 and k_2 in (15) in order to guarantee the finite time convergence of ${}^i\tilde{\boldsymbol{\zeta}}$ to the origin ($\forall\, i$). ■

In the rest of the paper, the vector $\tilde{\boldsymbol{\zeta}}^\star$, accounting for the collective reference estimation error and defined as

$$\tilde{\boldsymbol{\zeta}}^\star = \mathbf{1}_N \otimes \boldsymbol{\zeta}_d - \left[{}^1\boldsymbol{\zeta}^{\mathrm{T}}, \cdots, {}^N\boldsymbol{\zeta}^{\mathrm{T}}\right]^{\mathrm{T}}$$

will be adopted.

C. First layer. Global state observer

Let us introduce the vector

$$ {}^{i}\hat{\boldsymbol{x}} = \left[{}^{i}\hat{\boldsymbol{x}}_1^{\mathrm{T}}, {}^{i}\hat{\boldsymbol{x}}_2^{\mathrm{T}}, \ldots, \quad {}^{i}\hat{\boldsymbol{x}}_N^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{Np}, \tag{16} $$

where ${}^{i}\hat{\boldsymbol{x}}_j \in \mathbb{R}^p$ is the estimate made by arm i of $\boldsymbol{x}_j$ and ${}^{i}\hat{\boldsymbol{x}} \in \mathbb{R}^{Np}$ is the overall estimate of the system state $\boldsymbol{x}$ made by the ith arm. Moreover, the selection matrices $\boldsymbol{\Gamma}_i$ and $\boldsymbol{\Pi}_i$ are defined as

$$ \boldsymbol{\Gamma}_i = \{\boldsymbol{O}_p \quad \cdots \quad \underbrace{\boldsymbol{I}_p}_{i \text{ th node}} \quad \cdots \quad \boldsymbol{O}_p\} \in \mathbb{R}^{p\times Np}, $$

and

$$ \boldsymbol{\Pi}_i{=}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{\Gamma}_i{=}\mathrm{diag}\{\boldsymbol{O}_p \cdots \underbrace{\boldsymbol{I}_p}_{i \text{ th node}} \cdots \boldsymbol{O}_p\}{\in}\mathbb{R}^{Np\times Np}. $$

In order to have the ith robot estimate the system's state $\boldsymbol{x}$, the following update law was designed for ${}^{i}\hat{\boldsymbol{x}}$:

$$ {}^{i}\dot{\hat{\boldsymbol{x}}} = k_o\left(\sum_{j\in\mathcal{N}_i}\left({}^{j}\hat{\boldsymbol{x}} - {}^{i}\hat{\boldsymbol{x}}\right) + \boldsymbol{\Pi}_i\left(\boldsymbol{x} - {}^{i}\hat{\boldsymbol{x}}\right)\right) + {}^{i}\hat{\boldsymbol{u}}, \tag{17} $$

with

$$ {}^{i}\hat{\boldsymbol{u}}({}^{i}\hat{\boldsymbol{x}}, t) = \boldsymbol{J}_\sigma^{\dagger}\,{}^{i}\hat{\boldsymbol{\gamma}}{=}\boldsymbol{J}_\sigma^{\dagger}\left({}^{i}\hat{\boldsymbol{\zeta}}{-}k_\sigma\boldsymbol{\sigma}({}^{i}\hat{\boldsymbol{x}})\right) \tag{18} $$

where ${}^{i}\hat{\boldsymbol{\gamma}} = {}^{i}\hat{\boldsymbol{\zeta}} - k_\sigma\boldsymbol{\sigma}({}^{i}\hat{\boldsymbol{x}})$, $\boldsymbol{\sigma}({}^{i}\hat{\boldsymbol{x}}) = \boldsymbol{J}_\sigma{}^{i}\hat{\boldsymbol{x}}$ is the estimate of the cooperative task function made by robot i and k_o is a positive scalar gain to be designed.

Remark 4.2: In (17), it might seem that the unknown global state $\boldsymbol{x}$ is adopted by the ith robot. However, $\boldsymbol{\Pi}_i$ selects only ith component of the collective state $\boldsymbol{x}$, i.e., the robot's own state $\boldsymbol{x}_i$ that is supposed to be known from (2).

For the sake of notation compactness, the state estimates can be stacked into the vector

$$ \hat{\boldsymbol{x}}^{\star} = \left[{}^{1}\hat{\boldsymbol{x}}^{\mathrm{T}} \quad \ldots \quad {}^{N}\hat{\boldsymbol{x}}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{N^2p}, \tag{19} $$

and the collective estimation error is

$$ \begin{aligned} \tilde{\boldsymbol{x}}^{\star} &= \boldsymbol{1}_N{\otimes}\boldsymbol{x} - \hat{\boldsymbol{x}}^{\star} = \left[(\boldsymbol{x}{-}{}^{1}\hat{\boldsymbol{x}})^{\mathrm{T}},\ (\boldsymbol{x}{-}{}^{2}\hat{\boldsymbol{x}})^{\mathrm{T}},\ \ldots,(\boldsymbol{x}{-}{}^{N}\hat{\boldsymbol{x}})^{\mathrm{T}}\right]^{\mathrm{T}} \\ &= \left[{}^{1}\tilde{\boldsymbol{x}}^{\mathrm{T}},\ {}^{2}\tilde{\boldsymbol{x}}^{\mathrm{T}},\ \ldots,{}^{N}\tilde{\boldsymbol{x}}^{\mathrm{T}}\right]^{\mathrm{T}}. \end{aligned} \tag{20} $$

From (17), the collective estimation dynamics is given by

$$ \dot{\hat{\boldsymbol{x}}}^{\star} = -k_o\left(\boldsymbol{L}\otimes\boldsymbol{I}\right)\hat{\boldsymbol{x}}^{\star} + k_o\boldsymbol{\Pi}^{\star}\tilde{\boldsymbol{x}}^{\star} + \hat{\boldsymbol{u}}^{\star}, \tag{21} $$

where $\hat{\boldsymbol{u}}^{\star} = \left[{}^{1}\hat{\boldsymbol{u}}^{\mathrm{T}} \ \ldots \ {}^{N}\hat{\boldsymbol{u}}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{N^2p}$ and $\boldsymbol{\Pi}^{\star} = \mathrm{diag}\left\{\boldsymbol{\Pi}_1 \ \ldots \ \boldsymbol{\Pi}_N\right\} \in \mathbb{R}^{N^2p\times N^2p}$. The dynamics of the collective estimation error is

$$ \dot{\tilde{\boldsymbol{x}}}^{\star} = \boldsymbol{1}_N\otimes\dot{\boldsymbol{x}} - \dot{\hat{\boldsymbol{x}}}^{\star} = -k_o\tilde{\boldsymbol{L}}^{\star}\tilde{\boldsymbol{x}}^{\star} + \boldsymbol{1}_N\otimes\dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^{\star}, \tag{22} $$

with

$$ \tilde{\boldsymbol{L}}^{\star} = \boldsymbol{L}\otimes\boldsymbol{I}_{Np} + \boldsymbol{\Pi}^{\star} \in \mathbb{R}^{N^2p\times N^2p}, \tag{23} $$

and where the property $(\boldsymbol{L}\otimes\boldsymbol{I}_{Np})(\boldsymbol{1}_N\otimes\boldsymbol{x}) = \boldsymbol{0}_{N^2p}$ was exploited in (22).

Remark 4.3: In [4], it is shown that $-\tilde{\boldsymbol{L}}^{\star}$ in (23) is Hurwitz for directed strongly connected topologies. Morever, it is also symmetric for undirected graphs.

Remark 4.4: The designed strategy allows each manipulator to have an estimation of the end-effector configurations of all the manipulators in the team. Such a feature is required by the assumption on the nature of the cooperative task (13). However, differently from all the works cited in Section I and available in literature, the knowledge of this global information can potentially be used to achieve important secondary objectives as, for example, choosing the contact points in grasping tasks ensuring the grasp stability via the force-closure condition [24], [25] or to perform distributed fault detection and isolation [26].

D. Secon layer. Local adaptive control law

In this section the two main results of the paper are presented. Two cases are considered which are the pure-kinematic and the motion/force control. In both cases, let us consider the general control input $\boldsymbol{\tau}_i$ ($i = 1, 2, \ldots, N$) for system in (1) ([27]):

$$ \begin{aligned} \boldsymbol{\tau}_i &= \hat{\boldsymbol{M}}_i(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_{\sigma,i}{+}\hat{\boldsymbol{C}}_i(\boldsymbol{q}_i,\dot{\boldsymbol{q}}_i)\dot{\boldsymbol{q}}_{\sigma,i}{+}\hat{\boldsymbol{F}}_i\dot{\boldsymbol{q}}_{\sigma,i}{+}\hat{\boldsymbol{g}}(\boldsymbol{q}_i){+}k_q\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}{+}\boldsymbol{\Delta\tau}_i \\ &= \boldsymbol{Y}(\boldsymbol{q}_i,\dot{\boldsymbol{q}}_i,\dot{\boldsymbol{q}}_{\sigma,i},\ddot{\boldsymbol{q}}_{\sigma,i})\hat{\boldsymbol{\pi}}_i{+}k_q\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}{+}\ \boldsymbol{\Delta\tau}_i, \end{aligned} \tag{24} $$

where $\boldsymbol{\Delta\tau}_i \in \mathbb{R}^{n_i}$ is an additional input that will be exploited in the following and with the dynamic parameter update law:

$$ \dot{\hat{\boldsymbol{\pi}}}_i = \boldsymbol{K}_{\pi_i}^{-1}\boldsymbol{Y}_i^{\mathrm{T}}\dot{\tilde{\boldsymbol{q}}}_{\sigma,i} \tag{25} $$

where k_q is a positive scalar gain, $\boldsymbol{K}_{\pi_i} \in \mathbb{R}^{n_{\pi_i}\times n_{\pi_i}}$ is a symmetric positive definite gain matrix and

$$ \left\{\begin{array}{lcl} \dot{\boldsymbol{q}}_{\sigma,i} &=& \boldsymbol{J}_i^{\dagger}(\boldsymbol{\Gamma}_i\boldsymbol{J}_\sigma^{\dagger}\,{}^{i}\hat{\boldsymbol{\gamma}} + \boldsymbol{u}_{f,i}) + \dot{\boldsymbol{q}}_{n,i} \\ \ddot{\boldsymbol{q}}_{\sigma,i} &=& \boldsymbol{J}_i^{\dagger}(\boldsymbol{\Gamma}_i\boldsymbol{J}_\sigma^{\dagger}\,{}^{i}\dot{\hat{\boldsymbol{\gamma}}} + \dot{\boldsymbol{u}}_{f,i}) + \dot{\boldsymbol{J}}_i^{\dagger}\boldsymbol{J}_i\dot{\boldsymbol{q}}_{\sigma,i} + \ddot{\boldsymbol{q}}_{n,i} \\ \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} &=& \dot{\boldsymbol{q}}_{\sigma,i} - \dot{\boldsymbol{q}}_i \\ \ddot{\tilde{\boldsymbol{q}}}_{\sigma,i} &=& \ddot{\boldsymbol{q}}_{\sigma,i} - \ddot{\boldsymbol{q}}_i, \end{array}\right. \tag{26} $$

where the expression of $\ddot{\boldsymbol{q}}_{\sigma,i}$ comes by deriving the first of (26), by considering that $\boldsymbol{J}_\sigma$ is a constant matrix and that $\boldsymbol{J}_i\dot{\boldsymbol{q}}_{\sigma,i} = \boldsymbol{\Gamma}_i\boldsymbol{J}_\sigma^{\dagger}\,{}^{i}\hat{\boldsymbol{\gamma}}$. In addition:

- $\boldsymbol{J}_i^{\dagger} \in \mathbb{R}^{n_i\times p}$ is the pseudo-inverse matrix of the manipulator jacobian $\boldsymbol{J}_i$;
- $\dot{\boldsymbol{q}}_{n,i}$ is an additional joint velocity in the null space of $\boldsymbol{J}_i$ (i.e., $\boldsymbol{J}_i\dot{\boldsymbol{q}}_{n,i} = \boldsymbol{0}_p$) that can be used to avoid internal joint movements or satisfy secondary objectives as the avoidance of manipulator singularities (see Assumption 3.2) or obstacles present in the environment [28];
- $\boldsymbol{u}_{f,i} \in \mathbb{R}^p$ is an additional control input whose value is zero in the case of the pure kinematic control while it is different than zero in the case of interaction control and will be detailed later.

Moreover, Assumption 3.2 ensures that $\boldsymbol{J}_i^{\dagger}$ can always be computed in (26). Substitution of (24) in (1) leads to

$$ \begin{aligned} \boldsymbol{M}_i(\boldsymbol{q}_i)\ddot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \boldsymbol{C}_i(\boldsymbol{q}_i,\dot{\boldsymbol{q}}_i)\dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \boldsymbol{F}_i\dot{\tilde{\boldsymbol{q}}}_{\sigma,i} &= \\ \tilde{\boldsymbol{M}}(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_{\sigma,i}{+}\tilde{\boldsymbol{C}}_i(\boldsymbol{q}_i\dot{\boldsymbol{q}}_i)\dot{\boldsymbol{q}}_{\sigma,i}{+}\tilde{\boldsymbol{F}}_i\dot{\boldsymbol{q}}_{\sigma,i}{+}\tilde{\boldsymbol{g}}(\boldsymbol{q}_i){-}k_q\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}{+}\boldsymbol{J}_i^{\mathrm{T}}\boldsymbol{h}_i{-}\boldsymbol{\Delta\tau}_i &= \\ \boldsymbol{Y}_i(\boldsymbol{q}_i,\dot{\boldsymbol{q}}_i,\dot{\boldsymbol{q}}_{\sigma,i},\ddot{\boldsymbol{q}}_{\sigma,i})\tilde{\boldsymbol{\pi}}_i{-}k_q\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}{+}\boldsymbol{J}_i^{\mathrm{T}}\boldsymbol{h}_i{-}\boldsymbol{\Delta\tau}_i&, \end{aligned} \tag{27} $$

where $\tilde{\pi}_i = \pi_i - \hat{\pi}_i$ is the uncertainty on the i th manipulator dynamic parameters and

$$\begin{cases} \tilde{M}_i &= M_i - \hat{M}_i \\ \tilde{C}_i &= C_i - \hat{C}_i \\ \tilde{F}_i &= F_i - \hat{F}_i \\ \tilde{g}_i &= g_i - \hat{g}_i. \end{cases} \tag{28}$$

Finally, let us define the following vectors according to the notation in Table I and that will be useful in the following:

$$\dot{\tilde{q}}_\sigma = \left[\dot{\tilde{q}}_{\sigma,1}^{\mathrm{T}}, \ldots, \dot{\tilde{q}}_{\sigma,N}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^n \tag{29}$$

$$u_f = \left[u_{f,1}^{\mathrm{T}}, \ldots, u_{f,N}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{Np}. \tag{30}$$

V. Distributed pure-kinematic control

In the pure kinematic control case it is important to precisely control the end-effector formation. In this case, either manipulators are loosely coupled (as cooperative painting, gluing, etc.) or they might be adopted for the transportation or manipulation of deformable or flexible objects where it is more important to control their shape rather than the forces exerted on them. Moreover, the following theorem holds for this case.

Theorem 5.1: Let us consider model in (1) and the control input in (24) with $\Delta\tau_i = J_i^{\mathrm{T}} h_i$, $u_{f,i} = 0_p$, the dynamic parameters update law (25) and with the observer update law in (17), then, $\dot{\tilde{q}}_\sigma$ in (29), $\tilde{x}^\star$ and $\tilde{\sigma} = \sigma_d - \sigma$ are asymptotically convergent to the origin provided that k_o, k_σ and k_q are chosen such as

$$\begin{cases} k_o\lambda_L - \rho_1 > 0 \Leftrightarrow k_\sigma < \dfrac{k_o\lambda_L}{2N^2} \\ k_q > \dfrac{N\alpha^2}{4(k_o\lambda_L - \rho_1)}. \end{cases} \tag{31}$$

with

$$\begin{cases} \lambda_L &= \lambda_{min}(\tilde{L}^\star) & \\ \lambda_1 &= \lambda_{min}(F + k_q I_{Nn}) & > 0 \\ \rho_1 &= 2N^2 & > 0. \end{cases} \tag{32}$$

Proof: The proof is made of two parts that can be found in Appendix A and B. ■

In the theorem above, it is set $\Delta\tau = J_i^{\mathrm{T}} h_i$ and $u_{f,i} = 0_p$ in (24) and (26), respectively. Both conditions make manipulators infinitely rigid to interaction forces (as long as actuators are not in saturation) as required by the kinematic control case.

Remark 5.1: It must be remarked that the conditions on k_o in (31) is only sufficient, but it shows that the observer's dynamics is required to be faster than the controller's one. With regards to λ_L in (31), it seems that the tuning in (31) requires to known the communication topology in advance; however, for a given number of vehicles, the worst-case topology (i.e., the topology which minimizes λ_L) can be considered, thus achieving a somewhat conservative (but reliable) tuning.

The main components of the designed control scheme for such case are shown in Figure 1. In this figure, the output of the *Global Observer* in (17) is an estimation of the overall state of the system x; this estimation is, then, used by the *Global task control* in (18) to estimate the reference end-effectors' velocities. Finally, the latter are is used within a *Local adaptive control* law as in (24) and (25).

VI. Distributed interaction control

In the case of tasks that require a tight connection between manipulators and a grasped rigid object, a pure positional control might rise internal stresses and damage the object and/or the manipulators especially in the case where a central control unit is missing. Therefore, interaction generalized forces need to be explicitly taken into account. In particular, it is required that manipulators coordinate to manipulate/move a rigid object while regulating the squeezing forces on it. This is a problem that has been widely addressed in the last decades in the framework of centralized solutions as in [3] and references therein, but that raises several challenges, as detailed in the following, in the framework of distributed solutions and that has not been investigated so far. Based on [3], the collective generalized force vector h can be decomposed in a term contributing to the motion of the object $h_e \in \mathbb{R}^{Np}$ and in a term $h_{int} \in \mathbb{R}^{Np}$ that does not, representing the internal stresses

$$h = h_e + h_{int} = G^\dagger G h + (I_{Np} - G^\dagger G)h. \tag{33}$$

Based on equation (33), the contribution to the internal stresses of manipulator i can be computed from (33) as

$$\begin{aligned} h_{int,i} &= \Gamma_i h_{int} = h_i - \Gamma_i G^\dagger G h = h_i - \Gamma_i G^\dagger (G_i h_i + \sum_{j\neq i} G_j h_j) \\ &= \underbrace{(I_p - \Gamma_i G^\dagger G_i)h_i}_{local} \quad \underbrace{- \Gamma_i G^\dagger \sum_{j\neq i} G_j h_j}_{esternal}. \end{aligned} \tag{34}$$

The right-hand side of equation (34) shows that the contribution to the internal stresses by manipulator i on the object is the sum of a local contribute (known) and an external contribute (unknown in a decentralized framework) depending on the forces exerted by other manipulators and that needs to be locally estimated.

A. Distributed internal forces estimation

Because of Assumption 3.3, h_i in (34) is known, while $\sum_{j\neq i} G_j h_j$ is, as stated above, unknown in the framework at hand.
To this aim, from the point of view of the i th manipulator and considering (9), equation (8) can be rewritten as

$$M_o(x_o)\ddot{x}_o = G_i h_i + \sum_{j\neq i} G_j h_j - C_o(x_o, \dot{x}_o) - g_o(x_o) \tag{35}$$

where, in particular, $G_i h_i$ is the vector of generalized forces exerted by the i th manipulator and $\sum_{j\neq i} G_j h_j$ is the vector accounting for the generalized forces exerted by all the other manipulators. The estimation of the latter term to be used in (34) can be retrieved by manipulator i by the approach described in [29], and by defining vector $\theta_i(t) \in \mathbb{R}^p$ as

$$\theta_i(t) = K\left(\int_{t_0}^{t} (\alpha - G_i h_i - \theta_i)d\tau + m(t)\right), \tag{36}$$

where $K \in \mathbb{R}^{p\times p}$ is a constant diagonal positive definite matrix, $m(t) = M_o(x_o)\dot{x}_o$ is the generalized momentum of

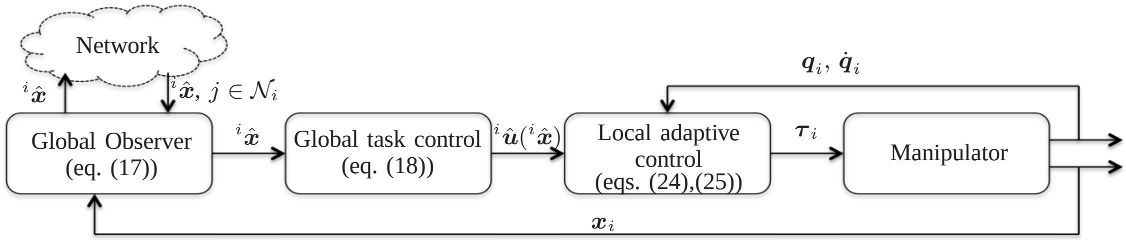

Fig. 1: The control scheme showing the main components of the designed solution with reference to the generic ith manipulator.

the object and $\boldsymbol{\alpha} = \boldsymbol{g}_o - \frac{1}{2}\dot{\boldsymbol{x}}_o^{\mathrm{T}} \frac{\partial \boldsymbol{M}_o}{\partial \boldsymbol{x}_o}\dot{\boldsymbol{x}}_o$.
In [29], it is shown that

$$\dot{\boldsymbol{\theta}}_i(t) = -\boldsymbol{K}\boldsymbol{\theta}_i(t) + \boldsymbol{K}\sum_{j\neq i}\boldsymbol{G}_j\boldsymbol{h}_j \tag{37}$$

which represents a low-pass filter that can be made arbitrarily fast by selecting high values of matrix $\boldsymbol{K} \in \mathbb{R}^{p\times p}$, asymptotically leading to $\boldsymbol{\theta}_i(t) \approx \sum_{j\neq i}\boldsymbol{G}_j\boldsymbol{h}_j$ and, from (34), to

$$\boldsymbol{h}_{int,i} = \boldsymbol{\Gamma}_i(\boldsymbol{I}_{Np} - \boldsymbol{G}^{\dagger}\boldsymbol{G})\boldsymbol{h} \approx (\boldsymbol{I}_p - \boldsymbol{\Gamma}_i\boldsymbol{G}^{\dagger}\boldsymbol{G}_i)\boldsymbol{h}_i - \boldsymbol{\Gamma}_i\boldsymbol{G}^{\dagger}\boldsymbol{\theta}_i. \tag{38}$$

Remark 6.1: In virtue of (34), the approximation error made in (38) about the computation of $\boldsymbol{h}_{int,i}$ is

$$\boldsymbol{h}_{int,i} - \Big((\boldsymbol{I}_p - \boldsymbol{\Gamma}_i\boldsymbol{G}^{\dagger}\boldsymbol{G}_i)\boldsymbol{h}_i - \boldsymbol{\Gamma}_i\boldsymbol{G}^{\dagger}\boldsymbol{\theta}_i\Big) = \boldsymbol{\Gamma}_i\boldsymbol{G}^{\dagger}(\boldsymbol{\theta}_i - \sum_{j\neq i}\boldsymbol{G}_j\boldsymbol{h}_j) \tag{39}$$

that is negligible only when the filter input (namely, $\sum_{j\neq i}\boldsymbol{G}_j\boldsymbol{h}_j$) has a bandwidth much smaller than the cut-off frequency of the filter. A practical choice is to set $\boldsymbol{K}$ in (36) as high as possible subject to the potential digital implementation of the filter itself.

It is worthy to say that, based on (38), the contribution of robot i to $\boldsymbol{h}_o$ (i.e., to the object motion) might be estimated by robot i as

$$\boldsymbol{h}_i - \boldsymbol{h}_{int,i} \approx \boldsymbol{\Gamma}_i(\boldsymbol{G}^{\dagger}\boldsymbol{G}_i\boldsymbol{h}_i + \boldsymbol{G}^{\dagger}\boldsymbol{\theta}_i) \tag{40}$$

that might be locally computed.

B. The control input

Therefore, in addition to the cooperative tracking of a motion task function $\boldsymbol{\sigma}_d$, it is necessary for each robot to regulate $\boldsymbol{h}_{int,i}$ to $\boldsymbol{h}^d_{int,i}$, $\forall i$. According to the notation adopted in the paper, the local force error $\tilde{\boldsymbol{h}}_{int,i} = \boldsymbol{h}^d_{int,i} - \boldsymbol{h}_{int,i}$ and the collective force error $\tilde{\boldsymbol{h}}_{int} = \begin{bmatrix}\tilde{\boldsymbol{h}}^{\mathrm{T}}_{int,1} & \dots & \tilde{\boldsymbol{h}}^{\mathrm{T}}_{int,N}\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{Np}$ are defined. The following theorem holds.

Theorem 6.1: Let us consider the model in (1) and the control input in (24) with the dynamic parameters update law (25) and with the observer update law in (17), then, $\dot{\tilde{\boldsymbol{q}}}_\sigma$, $\tilde{\boldsymbol{x}}^\star$ and $\tilde{\boldsymbol{h}}_{int}$ are asymptotically convergent to the origin provided that $\boldsymbol{u}_{f,i}$ in (26) is chosen as

$$\boldsymbol{u}_{f,i} = k_f\int_{t_0}^{t}\tilde{\boldsymbol{h}}_{int,i}d\tau \tag{41}$$

with $k_f > 0$, that k_o, k_q, k_σ, k_f are chosen such as

$$\begin{cases} k_o\lambda_L - \rho_1 > 0 \Leftrightarrow k_\sigma < \dfrac{k_o\lambda_L}{2N^2} \\ k_q > \dfrac{N\alpha^2}{4(k_o\lambda_L - \rho_1)} \\ k_f > \dfrac{\lambda_1}{4\left(\lambda_1(k_o\lambda_L - \rho_1 k_\sigma) - \alpha^2/4\right)}, \end{cases} \tag{42}$$

where parameters appearing in the system above are the same as in (32), and that $\Delta\boldsymbol{\tau}_i$ in (24) is set to

$$\Delta\boldsymbol{\tau}_i = \boldsymbol{J}_i^{\mathrm{T}}\Big(\boldsymbol{\Gamma}_i(\boldsymbol{G}^{\dagger}\boldsymbol{G}_i\boldsymbol{h}_i + \boldsymbol{G}^{\dagger}\boldsymbol{\theta}_i) + \boldsymbol{h}^d_{int,i} + k_f\boldsymbol{u}_{f,i}\Big) + \kappa_i(t)\dot{\tilde{\boldsymbol{q}}}_{\sigma,i} \tag{43}$$

where $\kappa_i(t)$ is a time-varying adaptive scalar gain satisfying the following relationship

$$\kappa_i(t) > \frac{\left\|\boldsymbol{\Gamma}_i\boldsymbol{J}_\sigma^{\dagger}\,{}^i\hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i\right\|\left\|\tilde{\boldsymbol{h}}_{int,i} + k_f\boldsymbol{u}_{f,i}\right\|}{\left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\right\|^2}. \tag{44}$$

Moreover, if $\boldsymbol{J}_\sigma\boldsymbol{u}_f = \boldsymbol{0}_m$ also $\tilde{\boldsymbol{\sigma}}$ converges to the origin.

Proof: The proof is made of two parts that can be found in Appendix A and C. ■

Remark 6.2: The additional control torque contribute $\Delta\boldsymbol{\tau}_i$ is made of three contributes:

- $\boldsymbol{\Gamma}_i(\boldsymbol{G}^{\dagger}\boldsymbol{G}_i\boldsymbol{h}_i + \boldsymbol{G}^{\dagger}\boldsymbol{\theta}_i)$ that represents, based on (40), the compensation of the contribute to the external generalized forces made by the ith manipulator on the object;
- $\boldsymbol{h}^d_{int,i} + k_f\boldsymbol{u}_{f,i}$ that represents a force feed-forward and integral error contribute;
- $\kappa_i(t)\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}$ that represents a robust term.

Remark 6.3: The term $\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\|$ is supposed to converge to zero and this might have $\kappa_i(t)$ grow unbounded. Therefore, the following approximation is made in practice

$$\begin{cases} \kappa_i(t) > \dfrac{\left\|\boldsymbol{\Gamma}_i\boldsymbol{J}_\sigma^{\dagger}\,{}^i\hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i\right\|\left\|\tilde{\boldsymbol{h}}_{int,i} + k_f\boldsymbol{u}_{f,i}\right\|}{\left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\right\|^2} & \text{if } \|\dot{\tilde{\boldsymbol{q}}}_\sigma\|^2 > \varepsilon \\ \kappa_i(t) = \dfrac{\left\|\boldsymbol{\Gamma}_i\boldsymbol{J}_\sigma^{\dagger}\,{}^i\hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i\right\|\left\|\tilde{\boldsymbol{h}}_{int,i} + k_f\boldsymbol{u}_{f,i}\right\|}{\varepsilon} & \text{if } \|\dot{\tilde{\boldsymbol{q}}}_\sigma\|^2 \leq \varepsilon \end{cases} \tag{45}$$

where ε is a positive scalar constant. This choice is common in robust control of uncertain systems and has as main consequence the ultimated boudedness of the solution with ultimate bound depending on the choice of the constant ε in [30],[31].

Remark 6.4: As stated by the theorem above, the adoption of a robust adaptive gain approach allows to have the internal forces converge to the origin despite the contact model and the compatibility of the motion and the force control tasks (i.e., $\boldsymbol{J}_\sigma \boldsymbol{u}_f$ might be different than zero). This choice has been made in order to make the internal force tracking the highest priority task, and avoid large internal stresses during the transient phase due to the initial observer and dynamic parameters errors.

The devised strategy is schematically represented in Figure 2. With respect to the scheme in Figure 1, the distributed internal force estimation and control blocks are added.

VII. Numerical simulations

In this section, the proposed solution is validated by simulation. Two cases studies are considered, the first one concerning distributed pure motion control while the second one concerning distributed motion/force control.

A. Distributed pure-kinematic control

In order to prove the effectiveness of the proposed approach, the case of 4 ($N=4$), 8-DOFs ($n_i = 8$, $i = 1,2,3,4$) Comau SmartSix serial chain manipulators (6-DOFs) mounted on a holonomic mobile base (2-DOFs) able to move in the $X-Y$ plane is considered (see Figure 4).
In this case study, ${}^i\boldsymbol{u}_f$ in (41) is set to zero ($\forall i$) as well as $\Delta\boldsymbol{\tau}_i$ in (43). The reduced regressor matrix has been computed such as $\boldsymbol{Y} = \boldsymbol{Y}_i \in \mathbb{R}^{6\times 56}$ ($\boldsymbol{\pi}_i \in \mathbb{R}^{56}$, $n_{\pi_i} = 56$, $i = 1,\,2,\,\ldots,\,4$), and its expression is not reported here due to its complex structure. The dynamic parameters $\boldsymbol{\pi}_i \in \mathbb{R}^{56}$, ($i = 1,\,2,\,\ldots,\,4$) are supposed to be known with an approximation of 15%. The end-effector configuration $\boldsymbol{x}_i \in \mathbb{R}^6$ ($p=6$) of the ith manipulator has the following components $\boldsymbol{x}_i = [p_{x,i},\ \ p_{y,i},\ \ p_{z,i},\ \ \phi_{1,i},\ \ \phi_{2,i},\ \ \phi_{3,i}]$, where the first three elements represent the end-effector position while the last three ones are a proper set of Euler-angles used to specify the end-effector orientation (ZYZ in our case). In order to simulate realistic conditions, a Gaussian random noise is added on the position and velocity measurements with zero mean and standard deviation equal to 0.02. In this case study, the objective is to cooperatively pick and place the blue object in the middle of the scene in Figure 4 from the position $(0,0,0)$ m to $(3,0,0)$ m, while keeping its orientation constant. To this aim, the 4 manipulators are required to approach, grasp, lift, move, lower and, finally, release the object with an overall duration of the motion of about 25 seconds.
In [32], it was shown that many tasks of practical importance as multi-arm object positioning or pure motion coordination can be easily described in the task-oriented framework by resorting to a proper set of *absolute-relative* variables. The position and orientation of the absolute frame is the centroid of the positions and orientations of the single tool frames, i.e.,

$$\boldsymbol{\sigma}_1 = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_i \; = \boldsymbol{J}_{\sigma,1}\boldsymbol{x}, \tag{46}$$

where $\boldsymbol{J}_{\sigma,1} = \frac{1}{4}\boldsymbol{1}_4^{\mathrm{T}} \otimes \boldsymbol{I}_6 \in \mathbb{R}^{6\times 24}$ is the task Jacobian matrix relative to the centroid task. The absolute variables are used to control the trajectory of the work-piece.

The relative motion between the arms can be described by the following task function

$$\boldsymbol{\sigma}_2 = \left[(\boldsymbol{x}_2-\boldsymbol{x}_1)^{\mathrm{T}},\ (\boldsymbol{x}_3-\boldsymbol{x}_2)^{\mathrm{T}},\ \ldots\ (\boldsymbol{x}_N-\boldsymbol{x}_{N-1})^{\mathrm{T}}\right]^{\mathrm{T}} = \boldsymbol{J}_{\sigma,2}\boldsymbol{x}, \tag{47}$$

where $\boldsymbol{J}_{\sigma,2}$ is the corresponding task Jacobian whose expression is

$$\boldsymbol{J}_{\sigma,2} = \begin{bmatrix} -\boldsymbol{I}_6 & \boldsymbol{I}_6 & \boldsymbol{O}_6 & \cdots & \boldsymbol{O}_6 \\ \boldsymbol{O}_6 & -\boldsymbol{I}_6 & \boldsymbol{I}_6 & \cdots & \boldsymbol{O}_6 \\ & & \vdots & & \\ \boldsymbol{O}_6 & \cdots & \boldsymbol{O}_6 & -\boldsymbol{I}_6 & \boldsymbol{I}_6 \end{bmatrix}. \tag{48}$$

The task vector

$$\boldsymbol{\sigma} = \begin{bmatrix}\boldsymbol{\sigma}_1\\ \boldsymbol{\sigma}_2\end{bmatrix} = \boldsymbol{J}_\sigma\boldsymbol{x} \in \mathbb{R}^{24},\ \text{with}\ \boldsymbol{J}_\sigma = \begin{bmatrix}\boldsymbol{J}_{\sigma,1}\\ \boldsymbol{J}_{\sigma,2}\end{bmatrix} \in \mathbb{R}^{24\times 24} \tag{49}$$

represents a useful set of task variables that can be used for grasping, carrying, deforming an object, etc., by properly assigning a desired trajectory to $\boldsymbol{\sigma}$, namely $\boldsymbol{\sigma}_d(t)$, [32], [33]. It is worth highlighting that in [32] a distributed planning strategy is designed in order to generate off-line joint trajectories, while this paper focuses on distributed control of uncertain mechanical systems in the operational space.

Moreover, since the selected mobile manipulators are redundant systems, the extra degrees of mobility can be used to achieve secondary objectives. In this case study, it is required to keep the joints of the serial chain manipulators far from their mechanical limits, i.e., $\dot{\boldsymbol{q}}_{n,i}$ in (26) has been chosen such as

$$\dot{\boldsymbol{q}}_{n,i} = (\boldsymbol{I}_8 - \boldsymbol{J}_i^{\dagger}\boldsymbol{J}_i)\dot{\boldsymbol{q}}_0 \ \text{with}\ \dot{\boldsymbol{q}}_0 = k_n \frac{\partial}{\partial \boldsymbol{q}} \sum_{l\in\mathcal{N}_{sc}} \frac{(q_l - \overline{q}_l)^2}{(q_{M,l} - q_{m,l})^2}; \tag{50}$$

were $\mathcal{N}_{sc}$ represents the index set of the joints relative to the serial chain manipulator, k_n is a scalar positive gain, $q_{m,l}$, $q_{M,l}$ and $\overline{q}_l$ are the minimum, maximum and the middle value of the i-th joint range, respectively.
The communication topology is shown in Figure 3, and it can be easily noticed that it is undirected and connected concerning the inter-robot communication, as required by Theorems 4.2. Moreover, as shown in the same figure it is assumed that the reference $\boldsymbol{\sigma}_d(t)$ is only known to robot 1 ($b_1 = 1$ and $b_j = 0$ for $j \neq i$ in (15)). Control gains are selected as: $k_o = 10$, $k_\sigma = 4$, $\boldsymbol{K}_\pi = 150\boldsymbol{I}_{56}$, $k_q = 80$ and $k_n = 5$ in (17), (26), (25), (32) and (50), respectively. In Figure 4, the manipulator and object's configurations in different time instants of the mission are shown. In the top left frame, manipulators approach the object and reached a configuration suitable for achieving object grasping. In the top right frame, manipulators approached the object along the horizontal plane and grasp the object. In the bottom left frame, manipulators and object are shown during the transportation. Finally, the object in the final configuration is shown in the bottom right frame.

In Figure 5 (top), the observer error $\|\tilde{\boldsymbol{x}}^{\star}\|$ is shown to asymptotically converge to zero as stated by Theorem 6.1. The initial observer error is not null since it is supposed that each arm does not know the initial configuration of the other ones (${}^i\tilde{\boldsymbol{x}}(t_0) \neq \boldsymbol{0}_{Np}$, $\forall\, i$). In the same figure, it is shown the task error $\tilde{\boldsymbol{\sigma}}$ (middle) and its time derivative (bottom). Also in this case, the task error asymptotically reaches the origin in accordance with Theorem 6.1.

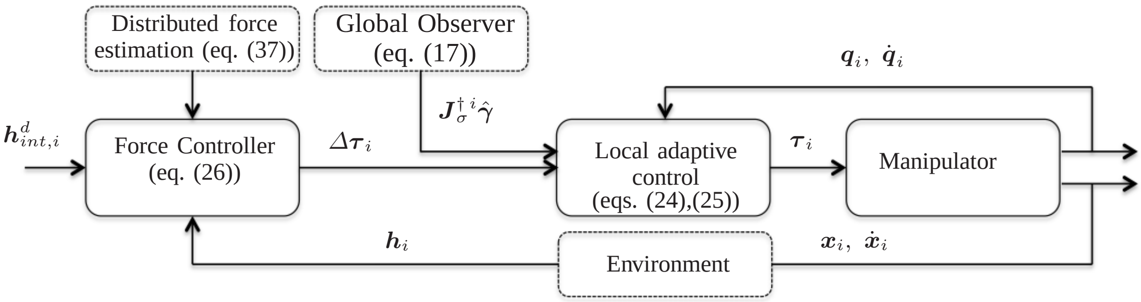

Fig. 2: The control scheme showing the main components of the designed solution in the perspective of cooperative distributed manipulation and with reference to the generic i th manipulator.

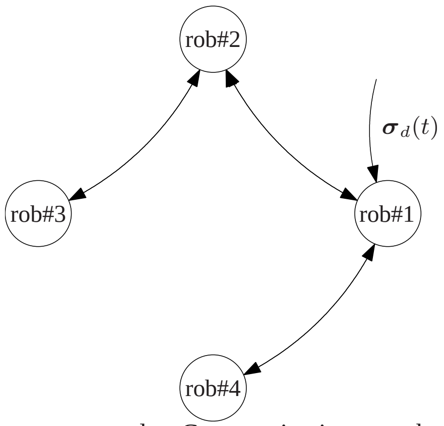

Fig. 3: First case study. Communication graph. The task reference $\boldsymbol{\sigma}_d$ and its derivatives are supposed to be known only by robot 1.

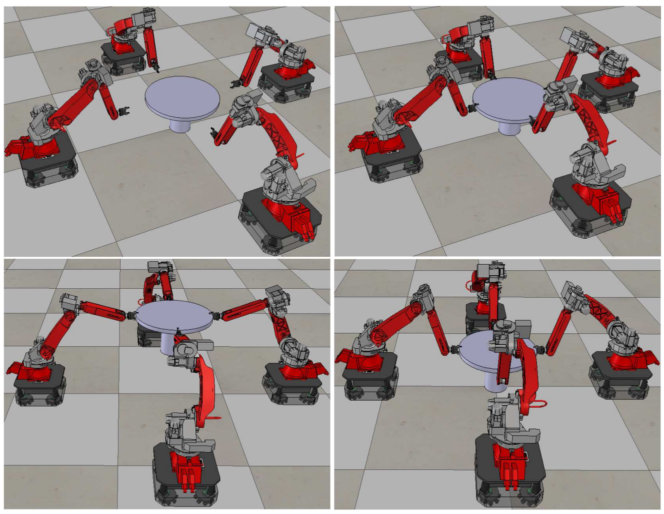

Fig. 4: First case study. Snapshots of robots' configurations during the object transportation. (Top Left) Manipulators have approached the object for grasping. (Top Right) Manipulators grasping the object and ready for transportation. (Bottom Left) Manipulators and object during the transportation. (Bottom right). The object in the final configuration.

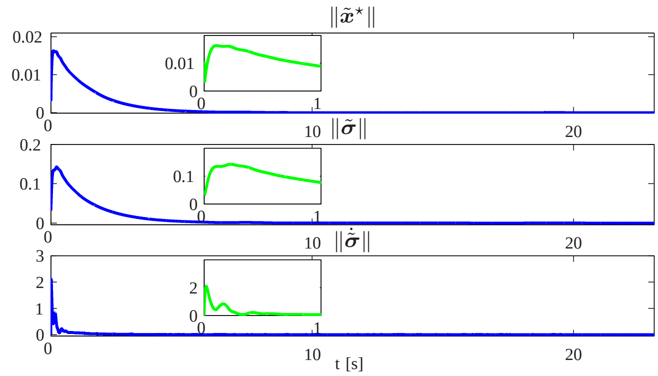

Fig. 5: First case study. (Top) Observer error $\|\tilde{\boldsymbol{x}}^{\star}\|$. (Middle) Task error $\|\tilde{\boldsymbol{\sigma}}\|$. (Bottom) Task velocity error $\|\dot{\tilde{\boldsymbol{\sigma}}}\|$.

In Figure 6, for the sake of completeness the torque inputs $\boldsymbol{\tau}_i$ relative to the 4 manipulator are shown.

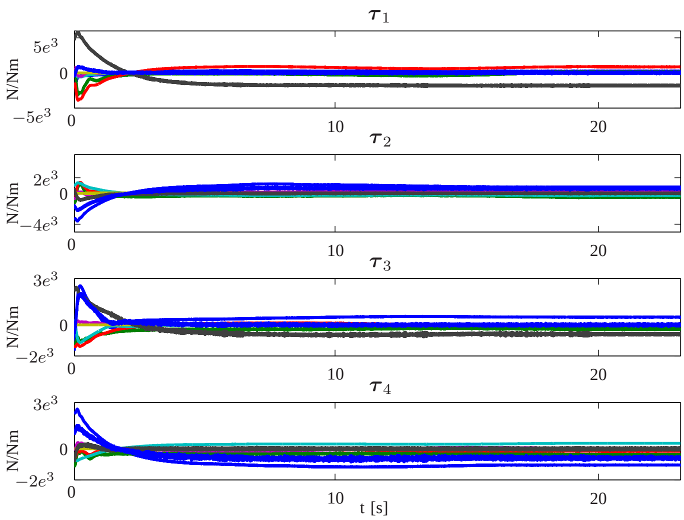

Fig. 6: First case study. Torque input on the 4 manipulators.

Remark 7.1: By choosing $\boldsymbol{\sigma} = \boldsymbol{x} = \boldsymbol{I}_{Nn}\boldsymbol{x} = \boldsymbol{J}_{\sigma}\boldsymbol{x}$ and $\boldsymbol{\sigma}_d(t) = \boldsymbol{1}_N \otimes \boldsymbol{a}_d(t)$, $\forall\ \boldsymbol{a}_d(t) \in \mathbb{R}^n$ is possible to solve the synchronization task to the value $\boldsymbol{a}_d(t)$. On the other hand, by choosing $\boldsymbol{\sigma} = \boldsymbol{\sigma}_2$ as in (47) and setting $\boldsymbol{\sigma}_d(t) = \boldsymbol{0}_m$ the rendez-vous task is achieved.

B. Distributed multi-robot interaction control

In order to show that pure distributed kinematic control is non effective in the case of tight connection of the manipulators through a common object, the internal forces and moments concerning the previous case study are shown in Figure 7. The simulation starts with the object already grasped by the object as in the top right frame in Figure 4. The object has a mass of $3\,$Kg and a tensor of inertia of $\boldsymbol{I}_o = \mathrm{diag}\{0.05,\, 0.05,\, 0.1\}\,\mathrm{m}^2\mathrm{Kg}$ with respect to its principal axes. The forces in Figure 7 are generated by considering a compliant contact between object and manipulators and simulating an error in the knowledge of the geometry of the object, with consequent error in the planning of the relative variables (i.e., $\boldsymbol{\sigma}_{2,d}$ in (47)) in particular along the x direction. This, together with the initial distributed observer and dynamic parameter errors, is the reason for large generalized internal forces in the transient phase in Figure 7. Instead, at steady state, the internal generalized forces reach a value of $-1800\,$N along the x direction in the case, for example, of manipulator 1. Therefore, it is required to control the generalized internal

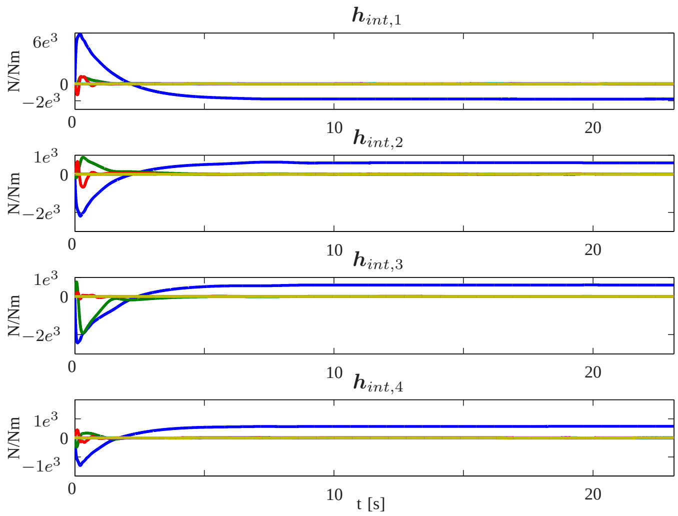

Fig. 7: First case study. Internal forces and moments relative to the 4 manipulators in the case of pure kinematic control. The blue component is relative to the x-axis.

forces. To this aim, the approach devised in Section VI is exploited.

In this case study, the same mobile manipulators as in the first case study are considered. The task $\boldsymbol{\sigma}(\boldsymbol{x})$ in (18) is assumed as $\boldsymbol{\sigma}_1(\boldsymbol{x})$ in (46), while the relative motion of the manipulators is exploited in order to control the internal stresses on the object. The communication topology with respect the inter-vehicles communication graph and the reference estimation input is the same as in Figure 3. Control gains k_o, k_σ, $\boldsymbol{K}_\pi$, k_q and k_n in (17), (26), (25), (32) and (50), respectively, are selected as in the previous case study, while k_f in (43) is selected as $k_f = 0.01$. Moreover, in addition to the noise on joint position and velocity measurements, also a zero mean Gaussian noise with 0.2 standard deviation was considered for force sensor measurements.

The force reference values $\boldsymbol{h}^d_{int,i}$ $(i = 1, 2, 3, 4)$ are set to

$$
\begin{aligned}
\boldsymbol{h}^d_{int,1} &= [7.5\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm}]^{\mathrm{T}}\\
\boldsymbol{h}^d_{int,2} &= [-2.5\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm}]^{\mathrm{T}}\\
\boldsymbol{h}^d_{int,3} &= [-2.5\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm}]^{\mathrm{T}}\\
\boldsymbol{h}^d_{int,4} &= [-2.5\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{N} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm} \quad 0\mathrm{Nm}]\,.^{\mathrm{T}}
\end{aligned}
$$

It is worth noticing that, based on (33), $\boldsymbol{h}^d_{int} \in \mathcal{N}\{\boldsymbol{G}\}$ and, then, it represents a vector of internal stresses to the object. Moreover, $\boldsymbol{J}_{\sigma,1} = \boldsymbol{J}_\sigma$ in (46) satisfies the motion/force compatibility condition $\boldsymbol{J}_\sigma \boldsymbol{h}^d_{int} = \boldsymbol{0}_m$ as required by Theorem 6.1.

Figure 8 shows the observer (top), task (middle) and task derivative (bottom) errors. As in the previous case study, these errors asymptotically reach the origin.

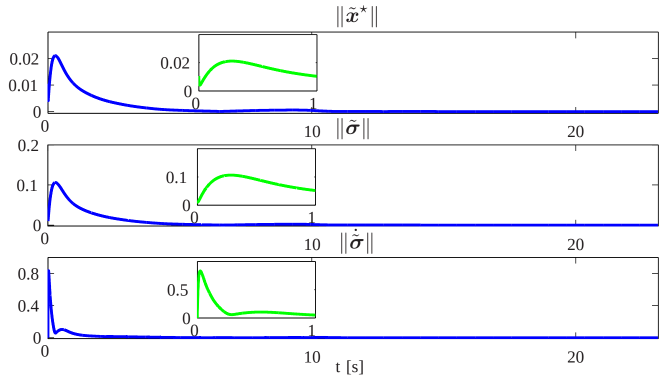

Fig. 8: Second case study. (Top) Observer error $\|\tilde{\boldsymbol{x}}^\star\|$. (Middle) Task error $\|\tilde{\boldsymbol{\sigma}}\|$. (Bottom) Task velocity error $\|\dot{\tilde{\boldsymbol{\sigma}}}\|$.

Concerning the internal forces and moments $\boldsymbol{h}_{int,i} \in \mathbb{R}^6$ adopted in (41), it is locally estimated by each mobile manipulator according to the algorithm in Section III-C, and the matrix gain $\boldsymbol{K}$ in (36) was set to $\boldsymbol{K} = 300\boldsymbol{I}_6$. Figure 9 reports the norm of the estimation error of $\boldsymbol{h}_{int,i}$ made by each robot and computed according to (39); this error is limited and asymptotically reaches the origin.

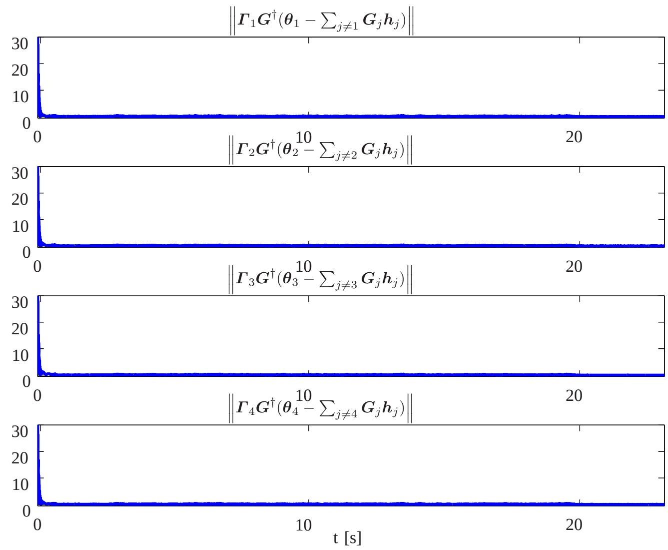

Fig. 9: Second case study. Norm of the estimation of the the internal forces $\boldsymbol{h}_{int,i}$ $(i = 1,\, 2,\, \ldots,\, 4)$ made by the filter in (36) relative to the 4 manipulators.

Figure 10 shows that the norm of the internal force tracking error (i.e., $\|\tilde{\boldsymbol{h}}_{int,i}\|$, $\forall\, i$) asymptotically reaches the origin and it is not affected by the initial observer error.

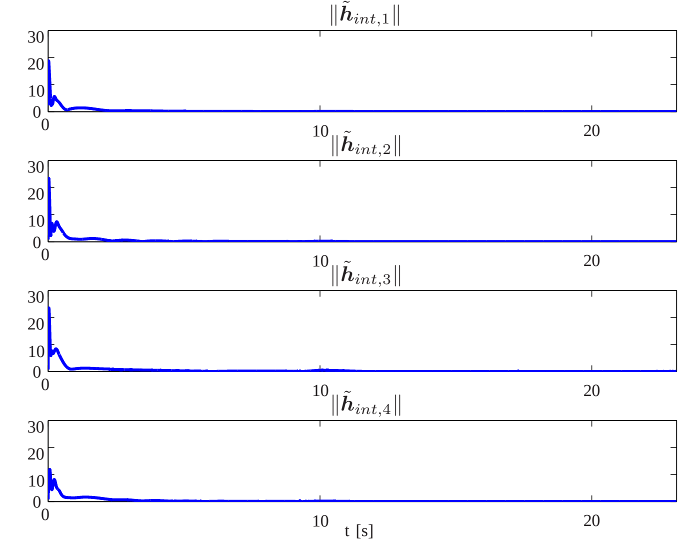

Fig. 10: Second case study. Norm of the internal forces and moments tracking errors relative to the 4 manipulators.

The measured internal forces are displayed in Figure 11. The figure shows that large peaks are avoided during the initial transient phase and even in presence of large observer errors; this is due to the adoption of the adaptive gain approach stated in Theorem 6.1.

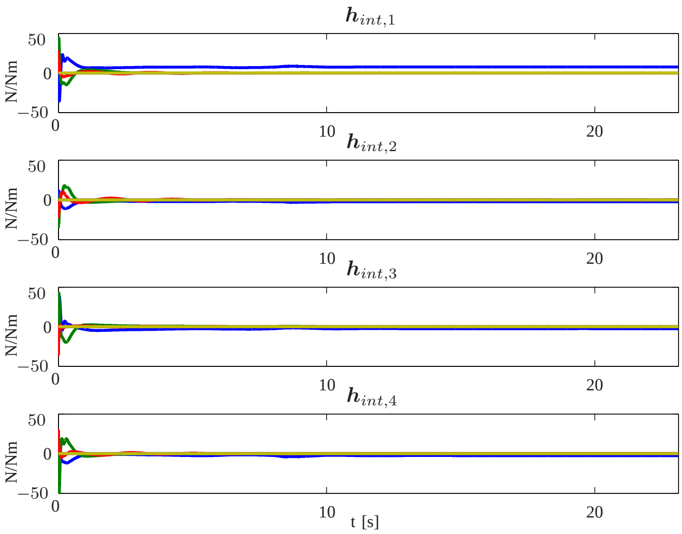

Fig. 11: Second case study. Internal forces and moments relative to the 4 manipulators in the case of interaction control.

Concerning $\kappa_i(t)$ in (44) ($\forall\, i$), the time history of the adaptive gains are shown in the Figure (12); the constant ε in (45) is set to $\varepsilon = 10^{-3}$. As it can be seen, the adaptive gains grow in the initial transient phase in order to counteract the initial observer error and the uncertainty on the robot dynamics, and asymptotically converges to zero as soon as the numerator in the expression of κ_i converges to zero.

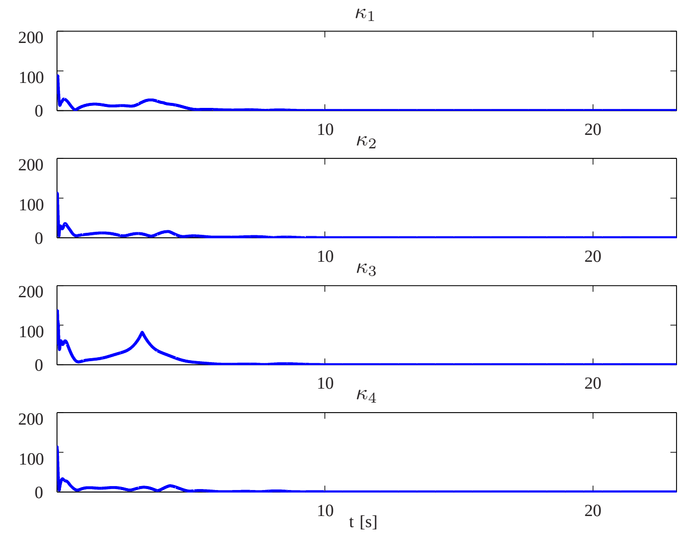

Fig. 12: Second case study. Adaptive gain $\kappa_i(t)$ in (45) ($\forall\, i$) with $\varepsilon = 10^{-3}$.

Finally, concerning joint generalized forces $\boldsymbol{\tau}_i$, they are presented for the sake of completeness in Figure 13.

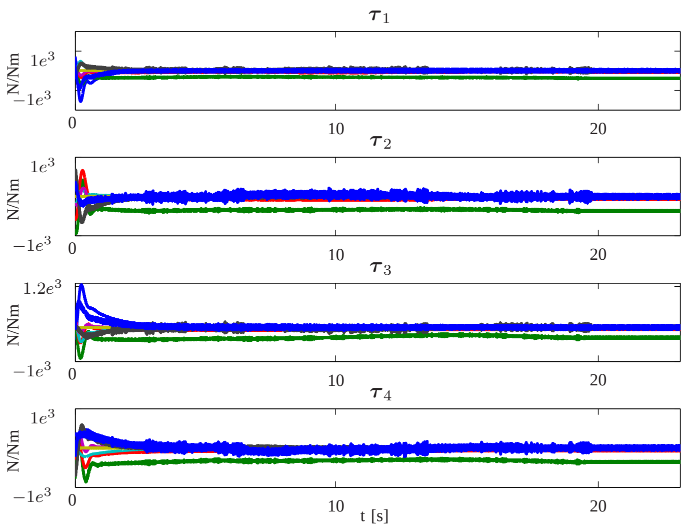

Fig. 13: Second case study. Torque input on the 4 manipulators.

VIII. Conclusions

A two layer architecture for cooperative control of Euler-Lagrange systems was designed. At the first level an observer/controller scheme is devised to estimate the full system state. At the second level, this estimate is used to compute an adaptive control law for uncertain systems. The aim is to achieve a global task function and the approach works for heterogeneous and redundant systems. Moreover, the devised solution is able to handle both pure kinematic and interaction control missions. The latter is particularly important in the case the robots are tightly connected via a stiff object. It is planned to validate the approach on a real setup.

APPENDIX

A. *Proof of Theorems 5.1 and 6.1*

The proof of both Theorems 5.1 and 6.1 is made of a common part which is represented by the two steps detailed in this section. Then, the proof will be differentiated in Sections B and C for the kinematic control and interaction control cases, respectively. In both cases, first it is proven the convergence to the origin of $\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}$ ($\forall\, i$) and $\tilde{\boldsymbol{x}}^\star$ for a proper choice of control gains. Based on this result, in the second step the convergence of $\boldsymbol{\sigma}_d - \boldsymbol{\sigma}$ to the origin is proved as well.

Step 1

Let us consider the following Lyapunov function

$$V = \sum_{i=1}^{N} V_{\tilde{q}_{\sigma,i}} + V_{\tilde{x}}, \tag{51}$$

where

$$V_{\tilde{q}_{\sigma,i}} = \frac{1}{2}\left(\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \boldsymbol{M}_i \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \tilde{\boldsymbol{\pi}}_i^{\mathrm{T}} \boldsymbol{K}_{\pi_i} \tilde{\boldsymbol{\pi}}_i\right)$$

and

$$V_{\tilde{x}} = \frac{1}{2} \tilde{\boldsymbol{x}}^{\star T} \tilde{\boldsymbol{x}}^\star.$$

By considering (22), the time derivative of V is

$$\begin{aligned}\dot{V} \;=\; & \sum_{i=1}^{N}\left(\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \boldsymbol{M}_i \ddot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \frac{1}{2}\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \dot{\boldsymbol{M}}_i \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \frac{1}{2}\tilde{\boldsymbol{\pi}}_i^{\mathrm{T}} \boldsymbol{K}_{\pi_i} \dot{\tilde{\boldsymbol{\pi}}}_i\right) \\ & - k_o \tilde{\boldsymbol{x}}^{\star T} \tilde{\boldsymbol{L}}^\star \tilde{\boldsymbol{x}}^\star + \tilde{\boldsymbol{x}}^{\star T}\left(\boldsymbol{1}_N \otimes \dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^\star\right),\end{aligned} \tag{52}$$

and by taking into account equation (27) and that matrix $\dot{\boldsymbol{M}}_i - 2\boldsymbol{C}_i$ is antisymmetric for a proper choice of matrix $\boldsymbol{C}_i$, equation (52) becomes

$$\begin{aligned}\dot{V} \;=\; & \sum_{i=1}^{N}\Big(-\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \boldsymbol{F}_i \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \boldsymbol{Y}_i \tilde{\boldsymbol{\pi}}_i - k_q \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \tilde{\boldsymbol{\pi}}_i^{\mathrm{T}} \boldsymbol{K}_{\pi_i} \dot{\tilde{\boldsymbol{\pi}}}_i \\ & \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}} \boldsymbol{h}_i - \boldsymbol{\Delta\tau}_i)\Big) - k_o \tilde{\boldsymbol{x}}^{\star T} \tilde{\boldsymbol{L}}^\star \tilde{\boldsymbol{x}}^\star + \tilde{\boldsymbol{x}}^{\star T}(\boldsymbol{1}_N \otimes \dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^\star) \\ =\; & \sum_{i=1}^{N}\Big(-\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \boldsymbol{F}_i \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + k_q \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \tilde{\boldsymbol{\pi}}_i^{\mathrm{T}}\Big(\boldsymbol{Y}_i^{\mathrm{T}} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} - \boldsymbol{K}_{\pi_i} \dot{\tilde{\boldsymbol{\pi}}}_i\Big) \\ & + \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}} \boldsymbol{h}_i - \boldsymbol{\Delta\tau}_i)\Big) - k_o \tilde{\boldsymbol{x}}^{\star T} \tilde{\boldsymbol{L}}^\star \tilde{\boldsymbol{x}}^\star + \tilde{\boldsymbol{x}}^{\star T}(\boldsymbol{1}_N \otimes \dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^\star)\end{aligned} \tag{53}$$

Finally, it is by choosing $\dot{\hat{\pi}}_i = \boldsymbol{K}_{\pi_i}^{-1} \boldsymbol{Y}_i^{\mathrm{T}} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}$ ($\forall i$) as in (25) and setting $\lambda_L = \lambda_{min}(\tilde{\boldsymbol{L}}^\star)$

$$\begin{aligned}\dot{V} \;\leq\; & -\sum_{i=1}^{N}\left(\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \boldsymbol{F}_i \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + k_q \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}} \boldsymbol{h}_i - \boldsymbol{\Delta\tau}_i)\right) \\ & - k_o \lambda_L \|\tilde{\boldsymbol{x}}^\star\|^2 + \tilde{\boldsymbol{x}}^{\star T}(\boldsymbol{1}_N \otimes \dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^\star).\end{aligned} \tag{54}$$

With regards to the term $(\boldsymbol{1}_N \otimes \dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^\star)$ in (54), it is

$$\begin{aligned}(\boldsymbol{1}_N \otimes \dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^\star) &= \begin{bmatrix} \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} - \boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}^\dagger(\boldsymbol{q})(\boldsymbol{J}_\sigma^{\dagger}\,{}^1\hat{\gamma}) \\ \vdots \\ \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} - \boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}^\dagger(\boldsymbol{q})(\boldsymbol{J}_\sigma^{\dagger}\,{}^N\hat{\gamma}) \end{bmatrix} \\ &= (\boldsymbol{I}_N \otimes \boldsymbol{J}(\boldsymbol{q}))\begin{bmatrix} \dot{\boldsymbol{q}} - \boldsymbol{J}^\dagger(\boldsymbol{q})(\boldsymbol{J}_\sigma^{\dagger}\,{}^1\hat{\gamma}) \\ \vdots \\ \dot{\boldsymbol{q}} - \boldsymbol{J}^\dagger(\boldsymbol{q})(\boldsymbol{J}_\sigma^{\dagger}\,{}^N\hat{\gamma}) \end{bmatrix}\end{aligned} \tag{55}$$

where $\boldsymbol{J}^\dagger = \mathrm{diag}\{\boldsymbol{J}_1^\dagger,\, \boldsymbol{J}_2^\dagger, \ldots, \boldsymbol{J}_N^\dagger\}$.
Because of the expression of ${}^i\hat{\gamma}$ in (18), the generic term $\boldsymbol{J}_\sigma^{\dagger}\,{}^i\hat{\gamma}$ can be expressed by some mathematical calculations as

$$\begin{aligned}\boldsymbol{J}_\sigma^{\dagger}\,{}^i\gamma &= \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\,{}^i\hat{\gamma} = \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}(\pm{}^k\hat{\gamma} + {}^i\hat{\gamma}) \\ &= \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\,{}^k\hat{\gamma} + \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}({}^i\hat{\gamma} - {}^k\hat{\gamma}) \\ &= \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\,{}^k\hat{\gamma} + \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\big[k_\sigma(-{}^i\hat{\boldsymbol{\sigma}} + {}^k\hat{\boldsymbol{\sigma}} \pm \boldsymbol{\sigma}) + {}^i\boldsymbol{\zeta} - {}^k\boldsymbol{\zeta} \pm \boldsymbol{\zeta}_d\big] \\ &= \begin{bmatrix} \boldsymbol{\Gamma}_1 \boldsymbol{J}_\sigma^{\dagger}\,{}^1\hat{\gamma} \\ \boldsymbol{\Gamma}_2 \boldsymbol{J}_\sigma^{\dagger}\,{}^2\hat{\gamma} \\ \vdots \\ \boldsymbol{\Gamma}_N \boldsymbol{J}_\sigma^{\dagger}\,{}^N\hat{\gamma} \end{bmatrix} + \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\Big[k_\sigma \boldsymbol{J}_\sigma({}^i\tilde{\boldsymbol{x}} - {}^k\tilde{\boldsymbol{x}}) - {}^i\tilde{\boldsymbol{\zeta}} + {}^k\tilde{\boldsymbol{\zeta}}\Big].\end{aligned} \tag{56}$$

By folding (56) in (55), the term $(\boldsymbol{1}_N \otimes \dot{\boldsymbol{x}} - \hat{\boldsymbol{u}}^\star)$ can be rewritten as in (57). Therefore, by folding (57) in (54) and by defining

$$\boldsymbol{F} = \mathrm{diag}\{\boldsymbol{F}_1,\, \boldsymbol{F}_2,\, \ldots,\, \boldsymbol{F}_n\},$$

it is

$$\begin{aligned}\dot{V} \leq\; & -\dot{\tilde{\boldsymbol{q}}}_\sigma^{\mathrm{T}}(\boldsymbol{F} + k_q \boldsymbol{I})\dot{\tilde{\boldsymbol{q}}}_\sigma - \tilde{\boldsymbol{x}}^{\star T}(\boldsymbol{I}_N \otimes \boldsymbol{J}(\boldsymbol{q}))(\boldsymbol{1}_N \otimes \dot{\tilde{\boldsymbol{q}}}_\sigma) - k_o \lambda_L \|\tilde{\boldsymbol{x}}^\star\|^2 \\ & - \tilde{\boldsymbol{x}}^{\star T}\begin{bmatrix} \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\left[k_\sigma \boldsymbol{J}_\sigma({}^1\tilde{\boldsymbol{x}} - {}^k\tilde{\boldsymbol{x}}) - {}^1\tilde{\boldsymbol{\zeta}} + {}^k\tilde{\boldsymbol{\zeta}}\right] \\ \vdots \\ \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\left[k_\sigma \boldsymbol{J}_\sigma({}^N\tilde{\boldsymbol{x}} - {}^k\tilde{\boldsymbol{x}}) - {}^N\tilde{\boldsymbol{\zeta}} + {}^k\tilde{\boldsymbol{\zeta}}\right] \end{bmatrix} \\ & + \tilde{\boldsymbol{x}}^{\star T}(\boldsymbol{1}_N \otimes \boldsymbol{u}_f) + \sum_{i=1}^{N} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}} \boldsymbol{h}_i - \boldsymbol{\Delta\tau}_i) \\ \leq\; & -\lambda_1\left\|\dot{\tilde{\boldsymbol{q}}}_\sigma\right\|^2 - k_o \lambda_L \|\tilde{\boldsymbol{x}}^\star\|^2 + \sqrt{N}\|\boldsymbol{I}_N \otimes \boldsymbol{J}(\boldsymbol{q})\|\left\|\dot{\tilde{\boldsymbol{q}}}_\sigma\right\|\|\tilde{\boldsymbol{x}}^\star\| \\ & + \|\tilde{\boldsymbol{x}}^\star\|\left\|\begin{bmatrix} \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\left[k_\sigma \boldsymbol{J}_\sigma({}^1\tilde{\boldsymbol{x}} - {}^k\tilde{\boldsymbol{x}}) - {}^1\tilde{\boldsymbol{\zeta}} + {}^k\tilde{\boldsymbol{\zeta}}\right] \\ \vdots \\ \sum_{k=1}^{N} \boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\left[k_\sigma \boldsymbol{J}_\sigma({}^N\tilde{\boldsymbol{x}} - {}^k\tilde{\boldsymbol{x}}) - {}^N\tilde{\boldsymbol{\zeta}} + {}^k\tilde{\boldsymbol{\zeta}}\right] \end{bmatrix}\right\| \\ & + \sqrt{N}\left\|\tilde{\boldsymbol{x}}^{\star T}\right\|\|\boldsymbol{u}_f\| + \sum_{i=1}^{N} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}} \boldsymbol{h}_i - \boldsymbol{\Delta\tau}_i) \\ \leq\; & -\lambda_1\left\|\dot{\tilde{\boldsymbol{q}}}_\sigma\right\|^2 - k_o \lambda_L \|\tilde{\boldsymbol{x}}^\star\|^2 + \alpha\sqrt{N}\left\|\dot{\tilde{\boldsymbol{q}}}_\sigma\right\|\|\tilde{\boldsymbol{x}}^\star\| \\ & + \|\tilde{\boldsymbol{x}}^\star\|\left(\sum_{i=1}^{N}\sum_{k=1}^{N}\left\|\boldsymbol{\Pi}_k \boldsymbol{J}_\sigma^{\dagger}\left(k_\sigma \boldsymbol{J}_\sigma({}^i\tilde{\boldsymbol{x}} - {}^k\tilde{\boldsymbol{x}}) - {}^i\tilde{\boldsymbol{\zeta}} + {}^k\tilde{\boldsymbol{\zeta}}\right)\right\|\right) \\ & + \sqrt{N}\|\tilde{\boldsymbol{x}}^\star\|\|\boldsymbol{u}_f\| + \sum_{i=1}^{N} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}} \boldsymbol{h}_i - \boldsymbol{\Delta\tau}_i) \\ \leq\; & -\lambda_1\left\|\dot{\tilde{\boldsymbol{q}}}_\sigma\right\|^2 - k_o \lambda_L \|\tilde{\boldsymbol{x}}^\star\|^2 + \alpha\sqrt{N}\left\|\dot{\tilde{\boldsymbol{q}}}_\sigma\right\|\|\tilde{\boldsymbol{x}}^\star\| \\ & \|\tilde{\boldsymbol{x}}^\star\|\left(\sum_{i=1}^{N}\sum_{k=1}^{N}\|\boldsymbol{\Pi}_k\|\left(\left\|\boldsymbol{J}_\sigma^{\dagger}\boldsymbol{J}_\sigma\right\|k_\sigma\left(\left\|{}^i\tilde{\boldsymbol{x}}\right\| + \left\|{}^k\tilde{\boldsymbol{x}}\right\|\right) + \left\|\boldsymbol{J}_\sigma^{\dagger}\right\|\left(\left\|{}^i\tilde{\boldsymbol{\zeta}}\right\| + \left\|{}^k\tilde{\boldsymbol{\zeta}}\right\|\right)\right)\right)\end{aligned}$$

$$
\begin{aligned}
(\mathbf{1}_N\otimes\dot{\boldsymbol{x}}-\hat{\boldsymbol{u}}^{\star}) &= (\boldsymbol{I}_N\otimes\boldsymbol{J}(\boldsymbol{q}))\left(\mathbf{1}_N\otimes\begin{bmatrix}\dot{\boldsymbol{q}}_1-\boldsymbol{J}_1^{\dagger}(\boldsymbol{\Gamma}_1\boldsymbol{J}_{\sigma}^{\dagger}\,{}^1\hat{\boldsymbol{\gamma}}\pm\boldsymbol{u}_{f,1})\\ \vdots\\ \dot{\boldsymbol{q}}_N-\boldsymbol{J}_N^{\dagger}(\boldsymbol{\Gamma}_N\boldsymbol{J}_{\sigma}^{\dagger}\,{}^N\hat{\boldsymbol{\gamma}}\pm\boldsymbol{u}_{f,N})\end{bmatrix}\right)-\begin{bmatrix}\sum_{k=1}^{N}\boldsymbol{\Pi}_k\boldsymbol{J}_{\sigma}^{\dagger}\left[k_{\sigma}\boldsymbol{J}_{\sigma}({}^1\tilde{\boldsymbol{x}}-{}^k\tilde{\boldsymbol{x}})+{}^1\tilde{\boldsymbol{\zeta}}-{}^k\tilde{\boldsymbol{\zeta}}\right]\\ \vdots\\ \sum_{k=1}^{N}\boldsymbol{\Pi}_k\boldsymbol{J}_{\sigma}^{\dagger}\left[k_{\sigma}\boldsymbol{J}_{\sigma}({}^N\tilde{\boldsymbol{x}}-{}^k\tilde{\boldsymbol{x}})+{}^N\tilde{\boldsymbol{\zeta}}-{}^k\tilde{\boldsymbol{\zeta}}\right]\end{bmatrix} \qquad (57)\\
&=-(\boldsymbol{I}_N\otimes\boldsymbol{J}(\boldsymbol{q}))\left(\mathbf{1}_N\otimes\begin{bmatrix}\dot{\tilde{\boldsymbol{q}}}_{\sigma,1}\\ \vdots\\ \dot{\tilde{\boldsymbol{q}}}_{\sigma,N}\end{bmatrix}\right)-\begin{bmatrix}\sum_{k=1}^{N}\boldsymbol{\Pi}_k\boldsymbol{J}_{\sigma}^{\dagger}\left[k_{\sigma}\boldsymbol{J}_{\sigma}({}^1\tilde{\boldsymbol{x}}-{}^k\tilde{\boldsymbol{x}})+{}^1\tilde{\boldsymbol{\zeta}}-{}^k\tilde{\boldsymbol{\zeta}}\right]\\ \vdots\\ \sum_{k=1}^{N}\boldsymbol{\Pi}_k\boldsymbol{J}_{\sigma}^{\dagger}\left[k_{\sigma}\boldsymbol{J}_{\sigma}({}^N\tilde{\boldsymbol{x}}-{}^k\tilde{\boldsymbol{x}})+{}^N\tilde{\boldsymbol{\zeta}}-{}^k\tilde{\boldsymbol{\zeta}}\right]\end{bmatrix}+\mathbf{1}_N\otimes\boldsymbol{u}_f=-(\boldsymbol{I}_N\otimes\boldsymbol{J}(\boldsymbol{q}))(\mathbf{1}_N\otimes\dot{\tilde{\boldsymbol{q}}}_{\sigma})-\begin{bmatrix}\sum_{k=1}^{N}\boldsymbol{\Pi}_k\boldsymbol{J}_{\sigma}^{\dagger}\left[k_{\sigma}\boldsymbol{J}_{\sigma}({}^1\tilde{\boldsymbol{x}}-{}^k\tilde{\boldsymbol{x}})+{}^1\tilde{\boldsymbol{\zeta}}-{}^k\tilde{\boldsymbol{\zeta}}\right]\\ \vdots\\ \sum_{k=1}^{N}\boldsymbol{\Pi}_k\boldsymbol{J}_{\sigma}^{\dagger}\left[k_{\sigma}\boldsymbol{J}_{\sigma}({}^N\tilde{\boldsymbol{x}}-{}^k\tilde{\boldsymbol{x}})+{}^N\tilde{\boldsymbol{\zeta}}-{}^k\tilde{\boldsymbol{\zeta}}\right]\end{bmatrix}+\mathbf{1}_N\otimes\boldsymbol{u}_f.
\end{aligned}
$$

$$
\begin{aligned}
&+\sqrt{N}\,\|\tilde{\boldsymbol{x}}^{\star}\|\|\boldsymbol{u}_f\|+\sum_{i=1}^{N}\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}}\boldsymbol{h}_i-\Delta\boldsymbol{\tau}_i)\\
\leq\;& -\lambda_1\left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\right\|^2-k_o\lambda_L\|\tilde{\boldsymbol{x}}^{\star}\|^2+\alpha\sqrt{N}\left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\right\|\|\tilde{\boldsymbol{x}}^{\star}\| \qquad (58)\\
&+\|\tilde{\boldsymbol{x}}^{\star}\|\left(\sum_{i=1}^{N}\sum_{k=1}^{N}\left\|\boldsymbol{J}_{\sigma}^{\dagger}\boldsymbol{J}_{\sigma}\right\|2k_{\sigma}\|\tilde{\boldsymbol{x}}^{\star}\|+2\left\|\boldsymbol{J}_{\sigma}^{\dagger}\right\|\left\|\tilde{\boldsymbol{\zeta}}^{\star}\right\|\right)\\
&+\sqrt{N}\,\|\tilde{\boldsymbol{x}}^{\star}\|\|\boldsymbol{u}_f\|+\sum_{i=1}^{N}\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}}\boldsymbol{h}_i-\Delta\boldsymbol{\tau}_i)\\
\leq\;& -\lambda_1\left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\right\|^2-k_o\lambda_L\|\tilde{\boldsymbol{x}}^{\star}\|^2+\alpha\sqrt{N}\left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\right\|\|\tilde{\boldsymbol{x}}^{\star}\|\\
&+2k_{\sigma}N^2\|\tilde{\boldsymbol{x}}^{\star}\|^2+2N^2\left\|\boldsymbol{J}_{\sigma}^{\dagger}\right\|\|\tilde{\boldsymbol{x}}^{\star}\|\left\|\tilde{\boldsymbol{\zeta}}^{\star}\right\|\\
&+\sqrt{N}\left\|\tilde{\boldsymbol{x}}^{\star T}\right\|\|\boldsymbol{u}_f\|+\sum_{i=1}^{N}\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}}\boldsymbol{h}_i-\Delta\boldsymbol{\tau}_i)\\
=\;& -\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\|\\ \|\tilde{\boldsymbol{x}}^{\star}\|\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2\\ -\alpha\sqrt{N}/2 & (k_o\lambda_L-\rho_1k\sigma)\end{bmatrix}\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\|\\ \|\tilde{\boldsymbol{x}}^{\star}\|\end{bmatrix}\\
&+\rho_1\left\|\boldsymbol{J}_{\sigma}^{\dagger}\right\|\|\tilde{\boldsymbol{x}}^{\star}\|\left\|\tilde{\boldsymbol{\zeta}}^{\star}\right\|+\sqrt{N}\,\|\tilde{\boldsymbol{x}}^{\star}\|\|\boldsymbol{u}_f\|+\sum_{i=1}^{N}\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}}(\boldsymbol{J}_i^{\mathrm{T}}\boldsymbol{h}_i-\Delta\boldsymbol{\tau}_i)
\end{aligned}
$$

with

$$
\begin{cases}\lambda_1 &= \lambda_{min}(\boldsymbol{F}+k_q\boldsymbol{I}_{Nn}) > 0\\ \rho_1 &= 2N^2 > 0.\end{cases} \qquad (59)
$$

Assumption 3.1 was exploited in (58) in setting $\|\boldsymbol{I}_N\otimes\boldsymbol{J}\|\leq\alpha$; moreover, it was also considered that $\|\boldsymbol{\Pi}_k\|=1$ $(\forall\, k)$ and that $\left\|\boldsymbol{J}_{\sigma}^{\dagger}\boldsymbol{J}_{\sigma}\right\|=1,\ \forall\,\boldsymbol{J}_{\sigma}$.

Step 2

Concerning the dynamics of $\boldsymbol{\sigma}_d-\boldsymbol{\sigma}$, let us notice that, based on the expression of $\boldsymbol{\gamma}$ in Table I and of its estimate $\boldsymbol{\gamma}_i$ in (18), it is

$$
\boldsymbol{\gamma}-{}^i\boldsymbol{\gamma}=\boldsymbol{\zeta}_d-k_{\sigma}\boldsymbol{\sigma}(\boldsymbol{x})-{}^i\hat{\boldsymbol{\zeta}}+k_{\sigma}\boldsymbol{\sigma}({}^i\hat{\boldsymbol{x}})=-k_{\sigma}\boldsymbol{J}_{\sigma}\,{}^i\tilde{\boldsymbol{x}}+{}^i\tilde{\boldsymbol{\zeta}}.
$$

By taking into account the above result, let us exploit the expression of $\dot{\tilde{\boldsymbol{q}}}_{\sigma}$:

$$
\dot{\tilde{\boldsymbol{q}}}_{\sigma}=\begin{bmatrix}\dot{\tilde{\boldsymbol{q}}}_{\sigma,1}\\ \vdots\\ \dot{\tilde{\boldsymbol{q}}}_{\sigma,N}\end{bmatrix}=\begin{bmatrix}\boldsymbol{J}_1^{\dagger}(\boldsymbol{\Gamma}_1\boldsymbol{J}_{\sigma}^{\dagger}({}^1\hat{\boldsymbol{\gamma}}\pm\boldsymbol{\gamma})+\boldsymbol{u}_{f,1})\\ \vdots\\ \boldsymbol{J}_N^{\dagger}(\boldsymbol{\Gamma}_N\boldsymbol{J}_{\sigma}^{\dagger}({}^N\hat{\boldsymbol{\gamma}}\pm\boldsymbol{\gamma})+\boldsymbol{u}_{f,N})\end{bmatrix}+\dot{\boldsymbol{q}}_n-\dot{\boldsymbol{q}}
$$

$$
=\boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{J}_{\sigma}^{\dagger}\boldsymbol{\gamma}+\boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{u}_f+\dot{\boldsymbol{q}}_n-\dot{\boldsymbol{q}}+\begin{bmatrix}\boldsymbol{J}_1^{\dagger}\boldsymbol{\Gamma}_1\boldsymbol{J}_{\sigma}^{\dagger}(k_{\sigma}\boldsymbol{J}_{\sigma}\,{}^1\tilde{\boldsymbol{x}}-{}^1\tilde{\boldsymbol{\zeta}})\\ \vdots\\ \boldsymbol{J}_N^{\dagger}\boldsymbol{\Gamma}_N\boldsymbol{J}_{\sigma}^{\dagger}(k_{\sigma}\boldsymbol{J}_{\sigma}\,{}^N\tilde{\boldsymbol{x}}-{}^N\tilde{\boldsymbol{\zeta}})\end{bmatrix} \qquad (60)
$$

with $\dot{\boldsymbol{q}}_n=\left[\dot{\boldsymbol{q}}_{n,1}^{\mathrm{T}},\ \dot{\boldsymbol{q}}_{n,2}^{\mathrm{T}},\ldots,\dot{\boldsymbol{q}}_{n,N}^{\mathrm{T}}\right]^{\mathrm{T}}$. Therefore, it holds

$$
\boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{J}_{\sigma}^{\dagger}\boldsymbol{\gamma}+\boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{u}_f+\dot{\boldsymbol{q}}_n-\dot{\boldsymbol{q}}=\dot{\tilde{\boldsymbol{q}}}_{\sigma}-\begin{bmatrix}\boldsymbol{J}_1^{\dagger}\boldsymbol{\Gamma}_1\boldsymbol{J}_{\sigma}^{\dagger}(k_{\sigma}\boldsymbol{J}_{\sigma}\,{}^1\tilde{\boldsymbol{x}}-{}^1\tilde{\boldsymbol{\zeta}})\\ \vdots\\ \boldsymbol{J}_N^{\dagger}\boldsymbol{\Gamma}_N\boldsymbol{J}_{\sigma}^{\dagger}(k_{\sigma}\boldsymbol{J}_{\sigma}\,{}^N\tilde{\boldsymbol{x}}-{}^N\tilde{\boldsymbol{\zeta}})\end{bmatrix} \qquad (61)
$$

By multiplying both members of (61) for matrix $\boldsymbol{J}(\boldsymbol{q})$ and since $\boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}_n=\mathbf{0}_{Np}$, it is

$$
\begin{aligned}
\boldsymbol{J}_{\sigma}^{\dagger}\boldsymbol{\gamma}-\boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}+\boldsymbol{u}_f &= \boldsymbol{J}_{\sigma}^{\dagger}\boldsymbol{\gamma}-\dot{\boldsymbol{x}}+\boldsymbol{u}_f\\
&= \boldsymbol{J}(\boldsymbol{q})\left(\dot{\tilde{\boldsymbol{q}}}_{\sigma}-\begin{bmatrix}\boldsymbol{J}_1^{\dagger}\boldsymbol{\Gamma}_1\boldsymbol{J}_{\sigma}^{\dagger}(k_{\sigma}\boldsymbol{J}_{\sigma}\,{}^1\tilde{\boldsymbol{x}}-{}^1\tilde{\boldsymbol{\zeta}})\\ \vdots\\ \boldsymbol{J}_N^{\dagger}\boldsymbol{\Gamma}_N\boldsymbol{J}_{\sigma}^{\dagger}(k_{\sigma}\boldsymbol{J}_{\sigma}\,{}^N\tilde{\boldsymbol{x}}-{}^N\tilde{\boldsymbol{\zeta}})\end{bmatrix}\right)=\boldsymbol{\xi}(\dot{\tilde{\boldsymbol{q}}}_{\sigma},\tilde{\boldsymbol{x}}^{\star},\tilde{\boldsymbol{\zeta}}),
\end{aligned} \qquad (62)
$$

where function $\boldsymbol{\xi}(\dot{\tilde{\boldsymbol{q}}}_{\sigma},\tilde{\boldsymbol{\sigma}}^{\star},\tilde{\boldsymbol{\zeta}})$ is equal to zero when its arguments are all equal to zero.

B. Kinematic control-Proof of Theorem 5.1

As from the statement of Theorem 5.1, it is set $\Delta\boldsymbol{\tau}=\boldsymbol{J}_i^{\mathrm{T}}\boldsymbol{h}_i$ and $\boldsymbol{u}_{f,i}=\mathbf{0}_p$ in (24) and (26), respectively. Therefore, (58) becomes in this case

$$
\begin{aligned}
\dot{V}\;\leq\;& -\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\|\\ \|\tilde{\boldsymbol{x}}^{\star}\|\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2\\ -\alpha\sqrt{N}/2 & (k_o\lambda_L-\rho_1k\sigma)\end{bmatrix}\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_{\sigma}\|\\ \|\tilde{\boldsymbol{x}}^{\star}\|\end{bmatrix}\\
&+\rho_1\left\|\boldsymbol{J}_{\sigma}^{\dagger}\right\|\|\tilde{\boldsymbol{x}}^{\star}\|\left\|\tilde{\boldsymbol{\zeta}}^{\star}\right\|.
\end{aligned} \qquad (63)
$$

In order for the quadratic form in (63) to be negative definite, it is necessary that the matrix

$$
\begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2\\ -\alpha\sqrt{N}/2 & k_o\lambda_L-\rho_1k_{\sigma}\end{bmatrix}
$$

is positive definite. This condition holds for

$$
\begin{cases} k_o\lambda_L-\rho_1>0\Leftrightarrow k_{\sigma}<\dfrac{k_o\lambda_L}{2N^2}\\ k_q>\dfrac{N\alpha^2}{4(k_o\lambda_L-\rho_1)},\end{cases} \qquad (64)
$$

whose a solution in the unknown k_o, k_q and k_σ always exists. Moreover, in virtue of Theorem 4.2, $\left\|\tilde{\boldsymbol{\zeta}}^\star\right\|$ reaches the origin in finite time T_s; then, $\dot{\tilde{\boldsymbol{q}}}_\sigma$ and $\tilde{\boldsymbol{x}}^\star$ exponentially converges to the origin.

Concerning $\boldsymbol{\sigma}_d - \boldsymbol{\sigma}$, by multiplying both members of (62) by $\boldsymbol{J}_\sigma$ and by considering that $\boldsymbol{u}_f = \boldsymbol{0}_{Np}$, it holds

$$\boldsymbol{\gamma} - \boldsymbol{J}_\sigma \dot{\boldsymbol{x}} = (\dot{\boldsymbol{\sigma}}_d - \dot{\boldsymbol{\sigma}}) + k_d(\boldsymbol{\sigma}_d - \boldsymbol{\sigma}) = \boldsymbol{J}_\sigma \boldsymbol{\xi}(\dot{\tilde{\boldsymbol{q}}}_\sigma, \tilde{\boldsymbol{x}}^\star, \tilde{\boldsymbol{\zeta}}). \quad (65)$$

Based on the previous stability result, the above equation represents an asymptotically stable system (in the state variable $\boldsymbol{\sigma}_d - \boldsymbol{\sigma}$) with vanishing input $\boldsymbol{J}_\sigma \boldsymbol{\xi}(\dot{\tilde{\boldsymbol{q}}}_\sigma, \tilde{\boldsymbol{x}}^\star, \tilde{\boldsymbol{\zeta}})$. Thus, $\boldsymbol{\sigma}_d - \boldsymbol{\sigma} = \boldsymbol{0}_m$ represents a globally asymptotically stable equilibrium [34]. This completes the proof.

C. Interaction control-Proof of Theorem 6.1

In this case, the Lyapunov function in (51) is modified as

$$V = \sum_{i=1}^N V_{\tilde{q}_{\sigma,i}} + \sum_{i=1}^N V_{h_i} + V_{\tilde{x}}, \quad (66)$$

where V_{h_i} is the term accounting for force error and is chosen as

$$V_{h_i} = \frac{1}{2k_f} \boldsymbol{u}_{f,i}^{\mathrm{T}} \boldsymbol{u}_{f,i}.$$

Based on the choice of $\boldsymbol{u}_{f,i}$ in (41) and of $\Delta\boldsymbol{\tau}_i$ in (43) and on (58), it is

$$\begin{aligned}
\dot{V} \le &-\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2 \\ -\alpha\sqrt{N}/2 & (k_o\lambda_L - \rho_1 k\sigma)\end{bmatrix} \begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix} \\
&+\rho_1 \left\|\boldsymbol{J}_\sigma^\dagger\right\| \|\tilde{\boldsymbol{x}}^\star\| \left\|\tilde{\boldsymbol{\zeta}}^\star\right\| + \sqrt{N}\|\tilde{\boldsymbol{x}}^\star\| \|\boldsymbol{u}_f\| \\
&+\sum_{i=1}^N \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} (\boldsymbol{J}_i^{\mathrm{T}} \boldsymbol{h}_i - \Delta\boldsymbol{\tau}_i) + \sum_{i=1}^N \boldsymbol{u}_{f,i}^{\mathrm{T}} \tilde{\boldsymbol{h}}_{int,i} \\
= &-\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2 \\ -\alpha\sqrt{N}/2 & (k_o\lambda_L - \rho_1 k\sigma)\end{bmatrix} \begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix} \\
&+\rho_1 \left\|\boldsymbol{J}_\sigma^\dagger\right\| \|\tilde{\boldsymbol{x}}^\star\| \left\|\tilde{\boldsymbol{\zeta}}^\star\right\| + \sqrt{N}\|\tilde{\boldsymbol{x}}^\star\| \|\boldsymbol{u}_f\| \\
&-\sum_{i=1}^N \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \boldsymbol{J}_i^{\mathrm{T}} \Big(\boldsymbol{\Gamma}_i(\boldsymbol{G}^\dagger \boldsymbol{G}_i \boldsymbol{h}_i + \boldsymbol{G}^\dagger \boldsymbol{\theta}_i) - \boldsymbol{h}_i + \boldsymbol{h}_{int,i}^d + k_f \boldsymbol{u}_{f,i}\Big) \\
&-\sum_{i=1}^N \kappa_i(t) \dot{\tilde{\boldsymbol{q}}}_{\sigma,i}^{\mathrm{T}} \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} + \sum_{i=1}^N \boldsymbol{u}_{f,i}^{\mathrm{T}} \tilde{\boldsymbol{h}}_{int,i}.
\end{aligned} \quad (67)$$

Since $\boldsymbol{J}_i \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} = (\boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^{\dagger\, i} \hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i + \boldsymbol{u}_{f,i})$, and in virtue of (40), (67) becomes

$$\begin{aligned}
\dot{V} \le &-\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2 \\ -\alpha\sqrt{N}/2 & (k_o\lambda_L - \rho_1 k\sigma)\end{bmatrix} \begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix} \\
&+\rho_1 \left\|\boldsymbol{J}_\sigma^\dagger\right\| \|\tilde{\boldsymbol{x}}^\star\| \left\|\tilde{\boldsymbol{\zeta}}^\star\right\| + \sqrt{N}\|\tilde{\boldsymbol{x}}^\star\| \|\boldsymbol{u}_f\| \\
&+\sum_{i=1}^N (\boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^{\dagger\, i} \hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i + \boldsymbol{u}_{f,i})^{\mathrm{T}} \left(-\tilde{\boldsymbol{h}}_{int,i} - k_f \boldsymbol{u}_{f,i}\right) \\
&-\sum_{i=1}^N \kappa_i(t) \left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\right\|^2 + \sum_{i=1}^N \boldsymbol{u}_{f,i}^{\mathrm{T}} \tilde{\boldsymbol{h}}_{int,i} \\
= &-\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2 \\ -\alpha\sqrt{N}/2 & (k_o\lambda_L - \rho_1 k\sigma)\end{bmatrix} \begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix} \\
&+\rho_1 \left\|\boldsymbol{J}_\sigma^\dagger\right\| \|\tilde{\boldsymbol{x}}^\star\| \left\|\tilde{\boldsymbol{\zeta}}^\star\right\| + \sqrt{N}\|\tilde{\boldsymbol{x}}^\star\| \|\boldsymbol{u}_f\| \\
&+\sum_{i=1}^N (\boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^{\dagger\, i} \hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i)^{\mathrm{T}} \left(-\tilde{\boldsymbol{h}}_{int,i} - k_f \boldsymbol{u}_{f,i}\right) \\
&-\sum_{i=1}^N \kappa_i(t) \left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\right\|^2 - \sum_{i=1}^N k_f \|\boldsymbol{u}_{f,i}\|^2 \\
\le &-\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2 \\ -\alpha\sqrt{N}/2 & (k_o\lambda_L - \rho_1 k\sigma)\end{bmatrix} \begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\|\end{bmatrix} \\
&+\rho_1 \left\|\boldsymbol{J}_\sigma^\dagger\right\| \|\tilde{\boldsymbol{x}}^\star\| \left\|\tilde{\boldsymbol{\zeta}}^\star\right\| + \sqrt{N}\|\tilde{\boldsymbol{x}}^\star\| \|\boldsymbol{u}_f\| \\
&+\sum_{i=1}^N \left\|\boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^{\dagger\, i} \hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i\right\| \left\|\tilde{\boldsymbol{h}}_{int,i} + k_f \boldsymbol{u}_{f,i}\right\| \\
&-\sum_{i=1}^N \kappa_i(t) \left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\right\|^2 - \sum_{i=1}^N k_f \|\boldsymbol{u}_{f,i}\|^2 \\
= &-\begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\| \\ \|\boldsymbol{u}_f\|\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\lambda_1 & -\alpha\sqrt{N}/2 & 0 \\ -\alpha\sqrt{N}/2 & (k_o\lambda_L - \rho_1 k\sigma) & -\sqrt{N}/2 \\ 0 & -\sqrt{N}/2 & k_f\end{bmatrix} \begin{bmatrix}\|\dot{\tilde{\boldsymbol{q}}}_\sigma\| \\ \|\tilde{\boldsymbol{x}}^\star\| \\ \|\boldsymbol{u}_f\|\end{bmatrix} \\
&-\sum_{i=1}^N \kappa_i(t) \left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\right\|^2 + \sum_{i=1}^N \left\|\boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^{\dagger\, i} \hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i\right\| \left\|\tilde{\boldsymbol{h}}_{int,i} + k_f \boldsymbol{u}_{f,i}\right\| \\
&+\rho_1 \left\|\boldsymbol{J}_\sigma^\dagger\right\| \|\tilde{\boldsymbol{x}}^\star\| \left\|\tilde{\boldsymbol{\zeta}}^\star\right\|.
\end{aligned} \quad (68)$$

By applying Sylvester's criterion, the matrix appearing in the right-hand side of the (68) is positive definite when

$$\begin{cases} k_o\lambda_L - \rho_1 > 0 \Leftrightarrow k_\sigma < \dfrac{k_o\lambda_L}{2N^2} \\ k_q > \dfrac{N\alpha^2}{4(k_o\lambda_L - \rho_1)} \\ k_f > \dfrac{\lambda_1}{4\left(\lambda_1(k_o\lambda_L - \rho_1 k_\sigma) - N\alpha^2/4\right)}, \end{cases} \quad (69)$$

whose a solution in the unknown k_o, k_q, k_σ and k_f always exists. Moreover, by locally choosing

$$\kappa_i(t) > \frac{\left\|\boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^{\dagger\, i} \hat{\boldsymbol{\gamma}} - \dot{\boldsymbol{x}}_i\right\| \left\|\tilde{\boldsymbol{h}}_{int,i} + k_f \boldsymbol{u}_{f,i}\right\|}{\left\|\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}\right\|^2}$$

and by considering that $\left\|\tilde{\boldsymbol{\zeta}}^\star\right\|$ reaches the origin in finite time, then $\dot{V}$ becomes semi-negative definite after T_s which, in turn, implies that V is bouded. By leveraging the boundedness of V and, then, of $\dot{\tilde{\boldsymbol{q}}}_{\sigma,i}$, $\tilde{\boldsymbol{\pi}}_i$, $\boldsymbol{u}_{f,i}$, and $\tilde{\boldsymbol{x}}^\star$ it can be easily shown the $\ddot{V}$ is bounded as well. Then, in virtue of Barbalat's lemma $\dot{V}$ is uniformly continuous and $\dot{V}$ converges to 0, as well as $\dot{\tilde{\boldsymbol{q}}}_\sigma$, $\tilde{\boldsymbol{x}}^\star$ and $\boldsymbol{u}_f = k_f \int_{t_0}^t \tilde{\boldsymbol{h}}_{int} d\tau$ (and $\tilde{\boldsymbol{h}}_{int}$).

Concerning $\boldsymbol{\sigma}_d - \boldsymbol{\sigma}$, by multiplying both members of (62) by $\boldsymbol{J}_\sigma$ as made before, it holds

$$\boldsymbol{\gamma} - \boldsymbol{J}_\sigma \dot{\boldsymbol{x}} - \boldsymbol{J}_\sigma \boldsymbol{u}_f = (\dot{\boldsymbol{\sigma}}_d - \dot{\boldsymbol{\sigma}}) + k_d(\boldsymbol{\sigma}_d - \boldsymbol{\sigma}) = \boldsymbol{J}_\sigma \boldsymbol{\xi}(\dot{\tilde{\boldsymbol{q}}}_\sigma, \tilde{\boldsymbol{x}}^\star, \tilde{\boldsymbol{\zeta}}). \tag{70}$$

In the case $\boldsymbol{J}_\sigma \boldsymbol{u}_f = \boldsymbol{0}_m$, based on the previous stability result, the above equation represents an asymptotically stable system (in the state variable $\boldsymbol{\sigma}_d - \boldsymbol{\sigma}$) with vanishing input $\boldsymbol{J}_\sigma \boldsymbol{\xi}(\dot{\tilde{\boldsymbol{q}}}_\sigma, \tilde{\boldsymbol{x}}^\star, \tilde{\boldsymbol{\zeta}})$ for which the same considerations made for (65) hold. This completes the proof.

REFERENCES

[1] D. Li S. Wang, J. Wan and C. Zhang. Implementing smart factory of industrie 4.0: An outlook. *International Journal of Distributed Sensor Networks*, 12(1), 2016.

[2] M.S. Essers andT. H. J. Vaneker. Design of a decentralized modular architecture for flexible and extensible production systems. *Mechatronics*, 30:160 – 169, 2015.

[3] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani. Six-dof impedance control of dual-arm cooperative manipulators. *IEEE/ASME Transactions on Mechatronics*, 13(5):576–586, Oct 2008.

[4] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. A decentralized controller-observer scheme for multi-agent weighted centroid tracking. *IEEE Trans. on Automatic Control*, 58(5):1310–1316, 2013.

[5] C. Soon-Jo and J.-J.E. Slotine. Cooperative robot control and concurrent synchronization of lagrangian systems. *IEEE Transactions on Robotics*, 25(3):686–700, June 2009.

[6] Shuai Li, Sanfeng Chen, Bo Liu, Yangming Li, and Yongsheng Liang. Decentralized kinematic control of a class of collaborative redundant manipulators via recurrent neural networks. *Neurocomputing*, 91:1–10, 2012.

[7] Y. C. Liu and N. Chopra. Controlled synchronization of heterogeneous robotic manipulators in the task space. *IEEE Transactions on Robotics*, 28(1):268–275, Feb 2012.

[8] Hanlei Wang. Task-space synchronization of networked robotic systems with uncertain kinematics and dynamics. *IEEE Transactions on Automatic Control*, 58(12):3169–3174, Dec 2013.

[9] F. Caccavale and M. Uchiyama. *Springer Handbook of Robotics*, chapter Cooperative Manipulation, pages 701–718. B. Siciliano, O. Khatib, (Eds.), Springer-Verlag, Heidelberg, D, 2008.

[10] G. Montemayor and J. T. Wen. Decentralized collaborative load transport by multiple robots. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 372–377, April 2005.

[11] Y. Hirata Z. Wang, Y. Takano and K. Kosuge. *Decentralized Cooperative Object Transportation by Multiple Mobile Robots with a Pushing Leader*, pages 453–462. Springer, 2007.

[12] Michael Rubenstein, Adrian Cabrera, Justin Werfel, Golnaz Habibi, James McLurkin, and Radhika Nagpal. Collective transport of complex objects by simple robots: theory and experiments. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 47–54. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[13] Anastasios Tsiamis, Christos K Verginis, Charalampos P Bechlioulis, and Kostas J Kyriakopoulos. Cooperative manipulation exploiting only implicit communication. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 864–869. IEEE, 2015.

[14] H. Bai and J. T. Wen. Cooperative load transport: A formation-control perspective. *IEEE Transactions on Robotics*, 26(4):742–750, August 2010.

[15] Z. Wang and M. Schwager. Kinematic multi-robot manipulation with no communication using force feedback. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 427–432, May 2016.

[16] T. Machado, T. Malheiro, S. Monteiro, W. Erlhagen, and E. Bicho. Multi-constrained joint transportation tasks by teams of autonomous mobile robots using a dynamical systems approach. In *IEEE International Conference on Robotics and Automation*, pages 3111–3117, May 2016.

[17] A. Petitti, A. Franchi, D. Di Paola, and A. Rizzo. Decentralized motion control for cooperative manipulation with a team of networked mobile manipulators. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 441–446, May 2016.

[18] A. Franchi, A. Petitti, and A. Rizzo. Decentralized parameter estimation and observation for cooperative mobile manipulation of an unknown load using noisy measurements. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5517–5522, May 2015.

[19] M. Mesbahi and M. Egerstedt. *Graph theoretic methods in multiagent networks*. Princeton Univiversity Press, 2010.

[20] F. Caccavale, P. Chiacchio, and S. Chiaverini. Task-space regulation of cooperative manipulators. *Automatica*, 36(6):879–887, June 2000.

[21] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. Decentralized time-varying formation control for multi-robot systems. *The International Journal of Robotics Research*, 33(7):1029–1043, 2014.

[22] G. Antonelli, C. Curatella, and A. Marino. Constrained motion planning for open-chain industrial robots. *Robotica, Cambridge*, 29(3):403–420, July 2011.

[23] G. Wen Y. Zhao, Z. Duan and Gu. Chen. Distributed finite-time tracking for a multi-agent system under a leader with bounded unknown acceleration. *Systems & Control Letters*, 81:8–13, 2015.

[24] R. Platt, A.H. Fagg, and R.A. Grupen. Null-space grasp control: Theory and experiments. *IEEE Trans. on Robotics*, 26(2):282–295, April 2010.

[25] Fabrizio Caccavale, Vincenzo Lippiello, Giuseppe Muscio, Francesco Pierri, Fabio Ruggiero, and Luigi Villani. Grasp planning and parallel control of a redundant dual-arm/hand manipulation system. *Robotica*, 31(07):1169–1194, 2013.

[26] F. Caccavale, A. Marino, G. Muscio, and F. Pierri. Discrete-time framework for fault diagnosis in robotic manipulators. *IEEE Transactions on Control Systems Technology*, 21(5):1858–1873, Sept 2013.

[27] J. J. E. Slotine and W. Li. *Applied nonlinear control*. Prentice-Hall, Englewood Cliffs, NJ, 1991.

[28] P. Hsu, J. Hauser, and S. Sastry. Dynamic control of redundant manipulators. In *1988 IEEE International Conference on Robotics and Automation*, pages 183–187 vol.1, Apr 1988.

[29] A. De Luca and R. Mattone. Sensorless robot collision detection and hybrid force/motion control. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 999–1004, April 2005.

[30] M. W. Spong. On the robust control of robot manipulators. *IEEE Transactions on Automatic Control*, 37(11):1782–1786, Nov. 1992.

[31] G. Leitmann. On the efficacy of nonlinear control in uncertain linear systems1. *Journal of Dynamic Systems, Measurement, and Control*, 103:95–102, Jan. 1981.

[32] F. Basile, F. Caccavale, P. Chiacchio, J. Coppola, and A. Marino. A decentralized kinematic control architecture for collaborative and cooperative multi-arm systems. *Mechatronics*, 23(8):1100–1112, 2013.

[33] F. Basile, F. Caccavale, P. Chiacchio, J. Coppola, and C. Curatella. Task-oriented motion planning for multi-arm robotic systems. *Robotics and Computer-Integrated Manufacturing*, 28:569–582, 2012.

[34] E. D. Sontag. Remarks on stabilization and input-to-state stability. In *IEEE Conference on Decision and Control*, pages 1376–1378, Dec. 1989.